Multi-layer Architecture and System Design of Internet Protocol (IP) and Optical Networks

Von der Fakultät für Elektrotechnik, Informationstechnik, Physik
der Technischen Universität Carolo-Wilhelmina zu Braunschweig

zur Erlangung der Würde

eines Doktor-Ingenieurs (Dr.-Ing.)

genehmigte Dissertation

von:	Mohit Chamania
aus:	Indore, India

eingereicht am:	26 Jul. 2013
mündliche Prüfung am:	14. Oct. 2013

Referent:	Mostafa Ammar, Professor, Georgia Tech
Referent:	Dr. Christoph Glingener, ADVA Optical Networking
Referent:	Admela Jukan, Professor, TU Braunschweig
Vorsitzender:	Harald Michalik, Professor, TU Braunschweig

Braunschweig, im Juli 2013

Abstract

The unprecedented growth in demand for Internet-based services in the last decade has driven innovations in networking to address issues of increasing capacity demand and infrastructure cost. At the same time, carrier-grade optical transport network architectures have evolved to emerge as a cost-effective alternative with support for dynamic provisioning of very-high bandwidth connections. As a result, researchers and operators have identified IP-optical integration as the key solution for operating the Internet infrastructure.

The modeling of multi-layer networks is well-researched problem, but existing models for IP-optical integration do not consider technology specific capabilities and operational aspects for employing dynamic optical circuits in IP networks. This thesis presents an ILP-based model which identifies and incorporates novel constraints for numerous technology specific aspects, such as IP forwarding capabilities and behavior of routing protocols. Novel solutions for critical operational aspects of IP-optical integration, such as optical circuit decommissioning and computation under unknown traffic conditions, are also proposed in this thesis.

The thesis identifies changes in routing as a major deterrent for employing dynamic optical circuits in IP networks, and proposes the new Optical Bypass approach to address the same. Quantitative studies presented indicate that the introduction of an optical circuit under this approach significantly reduces the effect on IP routing, while lowering optical capacity requirements as compared to the traditional Shortest Path First Routing based approaches. The proposed solution can also compute near-optimal solutions under unknown IP traffic matrix conditions, making it ideal for application in dynamic network scenarios.

The thesis also addresses specific management challenges with IP-optical integration, and outlines solutions to address the same. The solutions are built around enabling coordination of management subsystems in the two network layers. The thesis presents the general architecture to facilitate coordination between management subsystems in a programmable fashion and demonstrates the capability of the architecture to be used in legacy as well as SDN-capable infrastructure. The thesis also outlines the design and implementation of the first open-source PCE, which is a critical management subsystem for enabling multi-layer path computation in IP-optical networks.

Kurzfassung

Die im letzten Jahrzent stark gestiegene Nachfrage nach internetbasierten Dienstleistungen erforderte große Innovationen im Bereich der Telekommunikationsnetze, um dem zunehmenden Kapazitätsbedarf und den damit verbundenen Infrastrukturkosten gerecht zu werden. Gleichzeitig entwickelten sich optische Transportnetze zu einer kosteneffizienten Alternative, die zudem die dynamische Einrichtung von hochbitratigen Verbindungen (100Gbit/s und mehr) erlaubt. Infolgedessen betrachten Netzbetreiber und Forschungseinrichtungen die Integration der beiden Technologien IP (Paketvermittelung auf Schicht 3) und Optik (Transportnetze auf Schicht 2) als die Schlüssellösung für Betrieb und Management der Internet-Infrastruktur.

Grundsätzlich ist die Modellierung von Multilayer-Netzen ein bekanntes Problem, dennoch lassen die vorhandenen Modelle für IP-optische Integration viele technologiespezifische Eigenschaften und kritische Aspekte bei Einrichtung und Betrieb dynamischer optischer Verbindungen in IP-Netzwerken außer Acht. Hierzu gehören vor allem die Eigenheiten des Forwarding (Paket-Weiterleitung), sowie das Verhalten von Routingprotokollen. Die vorliegende Dissertation präsentiert ein auf linearer Optimierung basierendes Modell, dass solche Aspekte und Bedingungen identifiziert und integriert, sowie die Notwendigkeit hierfür anhand numerischer Evaluierung nachweist. Die berücksichtigten Modellierungsaspekte der IP-optischen Integration umfassen bezüglich des Netzbetriebs vor allem Probleme wie die optimale Platzierung dynamischer optischer Verbindungen im Falle einer unbekannten Verkehrsmatrix sowie deren Verbindungsabbau.

Ein wesentliches Hindernis zur Verwendung dynamischer optischer Verbindungen in IP-Netzen sind die aus ihrer Einrichtung resultierenden protokollspezifischen Änderungen im Routing. Als Lösung wird in dieser Dissertation der sogenannte optische Bypass vorgeschlagen. Die hier gezeigten Untersuchungen zeigen, dass der Einsatz optischer Bypässe die Auswirkungen auf das IP-Routing stark reduziert und gleichzeitig die dafür notwendigen Kapazitätsanforderungen im

optischen Transportnetz verringert. Damit bieten optische Bypässe sich in dynamischen Netzwerkszenarien an und sind ebenfalls für Anwendungsszenarien geeignet, in denen die IP-Verkehrsmatrix nicht bekannt ist.

Im Weiteren werden noch Probleme aus dem Bereich Netzwerkmanagement behandelt, die sich spezifisch aus der IP-optischen Integration entwickeln, und es werden entsprechende Lösungsansätze vorgestellt. Diese basieren im Wesentlichen auf der Abstimmung und dem koordinierten Betrieb der Management-Subsysteme in den beiden betroffenen Netzschichten. Gezeigt wird dann eine allgemeine Architektur, die eine Koordination von Management-Subsystemen in programmierbarer Form ermöglicht. Insbesondere wird demonstriert, dass diese Architektur sowohl für Standardtechnologien als auch für SDN-Infrastruktur (Software Defined Networking) geeignet ist. Außerdem wird der Entwurf und die Implementierung des Open Source PCE (Path Computation Element) beschrieben, welches als entscheidendes Management-Subsystem die Berechnung von Multilayer Verbindungswegen in IP-optischen Netzwerken ermöglicht.

Acknowledgments

This thesis work would not have been possible without the inspiration and support of many people. I would like to start by expressing my heartfelt thanks to my PhD advisor, Prof. Admela Jukan. Her unbridled enthusiasm and commitment to her work have been the driving force for during the challenging times in this journey. She has been a strong and supportive advisor and was the primary sounding board for the ideas presented in this thesis. She also gave me the freedom and resources to lead independent research activities within her group, and showed enormous faith in letting me lead research activities within European research projects. Her dedication, and the never-ending capacity to go the extra mile for her students day-in and day-out have been inspiring, and for that I shall be forever grateful. I would also like to thank the members of my thesis committee, Prof. Mostafa Ammar and Dr. Christoph Glingener for their valuable time to evaluate my work. I am also grateful to Prof. Harald Michalik who generously accepted to chair the examination committee.

I was extremely fortunate to have some wonderful people as colleagues over the course of my thesis. A special thanks to Xiaomin Chen, Marcel Caria and Marek Drogon who have been excellent collaborators over the course of my thesis. Working with them has been a pleasure, and their friendship has been invaluable. I am also very grateful to Dr. Wolfgang Bziuk, Dr. Tamal Das, Dr. Silvana Greco and Dr. Said Zaghloul, who have been supportive colleagues and great friends. I would also like to thank the partners of the ONE project and the 100 GET project who have been excellent collaborators. A special thanks to our administrative staff, especially Ms. Ruth Kiepert, Ms. Christiane Geller and Ms. Nadine Becker for their support during the course of this thesis.

Special thanks also go to my former professors at IIT Bombay for the remarkable learning experience during the course of my Bachelors and Masters studies. I would like to especially thank Prof. Ashwin Gumaste, who introduced me to Optical Networking and

gave me the opportunity to work in his research group.

None of this would have been possible without the love and support of my family and friends. My parents and sister have always been a constant pillar of support in all my pursuits, and I could not have finished this without them. My friends also have been a constant source of encouragement throughout this endeavor, and for that and more, I shall be forever grateful.

List of Figures

2.1 Generic IP-over-Optical Network Scenario used to model the network 22
2.2 Routing Continuity in IP networks 32
2.3 Example depicting destination-based forwarding in IP networks 34
2.4 Example describing the need for the routing re-convergence constraints in Shortest Path First Routing (SPF) formulation 37
2.5 Traffic measurements available in IP networks [1] . . 42

3.1 Typical Deployment of IP-Optical networks with dynamic optical circuits (Based on the LambdaStation [2] Deployment scenario) 50
3.2 Example describing the use of Optical Bypass in a network 51
3.3 Routing Constants and Variables used for formulating the Optical Bypass Model 54
3.4 Performance of different schemes on the NSFNet network for different traffic overloading scenarios: (Scenario 1) Traffic on randomly selected router pairs is increased by 150%; (Scenario 2) Traffic on all router pairs is increased by a factor as indicated. Error bars indicate 95% confidence intervals. (Confidence intervals on the results for routing changes were very small and not been shown in the figures for clarity.) [1] . . 63

3.5 Optical circuits as new links or capacity boost for existing links when traffic of 17-19 s-d pairs in NSFNet is increased by 150% [1] 66
3.6 Difference in installed capacity with larger N when traffic on 17-19 s-d pairs in NSFNet is increased by 150% [1] 67
3.7 Performance of different mechanisms in the 14-node ring and the 4x4 grid (mesh) topology when overloading is achieved by increasing the traffic on a number (as indicated on the X-axis) of randomly selected source-destination pairs by 150 % [1] 69
3.8 Average installed capacity at each step for switch-off in the NSFNet topology. [1] 73
3.9 Average routing configurations at each step for switch-off in the NSFNet topology. [1] 74

4.1 The *Module* interface indicating the interfaces exposed by all modules [3] 84
4.2 *Modular* architecture of the PCE and the PCC indicating the interactions between the different modules [3] 85
4.3 Object hierarchy of the PCEP message implementation [3] 88
4.4 The implementation of the Network Module with Asynchronous Network I/O [3] 89
4.5 The PCEP state machine and associated PCEP message exchange between PCE peers [3] 91
4.6 Implementation of the Session Management Module [3] 92
4.7 Implementation of the Computation Module [3] . . . 93
4.8 Indication of the different processing time measurements made for the PCE scalability tests [3] 95

4.9 Measured processing time for path computation for different algorithms and topology sized [3] 97
4.10 Effect of TED update frequency on processing time in the computation module [3] 100
4.11 Automated Framework for Multi-layer connection provisioning [4] 104
4.12 The multi-layer testbed used to demonstrate multi-layer path provisioning [4] 106
4.13 Wireshark snapshots of the signaling for path computation between the IP NCM, the MPLS PCE and the WSON PCE [4] 109
4.14 UNI PATH and RESV messages for optical circuit setup in the WSON network [4] 111

5.1 Overview of the ONE adapter Architecture [5] . . . 120
5.2 Architecture of the Workflow Processor [6] 122
5.3 Architecture of the Trigger Module [6] 124
5.4 Communication between the ONE adapter modules using the Enterprise Service Bus architecture [5] . . 130
5.5 Example of IP Traffic offloading using the ONE adapter 133

List of Tables

3.1 Interface types and associated parameters used for the numerical evaluation 61

3.2 Optical capacity required under limited combination of routing and forwarding constraints [1] 68

3.3 Computation times (seconds) in the NSFNet topology [1] 75

4.1 Time measurements (in nanoseconds) for different processes inside modules in the PCE server (shown in Fig. 4.8). Load indicates number of PCE client instances with each instance generating requests using a Poisson distribution (average inter-arrival time = 1 sec) [3] 96

4.2 Measured processing times (in nanoseconds) in the request queue and for path computation when varying thread pool size for the maximum bandwidth shortest path algorithm applied to the 200 node network topology. Requests are generated using a Poisson process with the mean inter-arrival time = 400 ms. [3] . . . 98

Contents

1 Introduction 1

1.1 Thesis Contributions 5

1.1.1 Modeling Effects of Dynamic Optical Circuit Provisioning on IP Routing 6

1.1.2 Optical Bypass - A Novel Mechanism for Use of Dynamic Optical Circuits in IP Networks . 7

1.1.3 Systems and Standards Challenges Associated with Coordinated Multi-layer Path Computation 8

1.1.4 Coordinated Orchestration of Management Operations Between IP and Transport Network Management Systems (NMS)s 9

1.2 Supporting Publications 10

1.2.1 Journal Articles 10

1.2.2 Conferences and Workshops 11

1.3 Thesis Organization 14

2 Modeling IP-Optical Integration Under Traditional IP Routing 17

2.1 Supporting Publications 18

2.2 Limitations of Existing IP-Optical Modeling Solutions 19

2.3 Network Model Description 21

2.3.1 Network Model Assumptions 21

2.3.2 Network Architecture Formulation and Constraints 24

2.3.3 Optimization Objective 26

2.4 Routing in IP Networks 30
2.4.1 Explicit Routing (ER) 31
2.4.2 Explicit Routing Under Destination-based Forwarding (ED-D) 33
2.4.3 Shortest Path First Routing (SPF) 34
2.5 Routing of Dynamic Optical Circuits 39
2.6 Formulation Under Unknown Traffic Matrix Conditions 41
2.7 Summary 45

3 New Approach for IP-Optical Integration - Optical Bypass 47
3.1 Supporting Publications 48
3.2 Motivation 49
3.3 The Optical Bypass (BY) Approach 50
3.4 Formulation of the Optical Bypass 52
3.4.1 Problem Formulation - Optical Bypass (BY) 53
3.4.2 Problem Formulation - Optical Bypass with Destination-Based Forwarding 57
3.4.3 Formulation Under Unknown Traffic Matrix Conditions 58
3.5 Numerical Evaluation 59
3.5.1 Performance Under Different Traffic Conditions 62
3.5.2 Performance in Different Topologies 68
3.5.3 Effects of Dynamic Optical Circuit Decommissioning 72
3.5.4 Computation Complexity 75
3.6 Summary 76

4 Multi-layer PCE-based Service Provisioning 79
4.1 Supporting Publications 80
4.2 Overview of the Internet Engineering Task Force (IETF) Path Computation Element (PCE) Architecture . . 81
4.3 Open Source PCE 82
4.3.1 Architecture and Implementation Specifications 84

4.3.2 PCE Performance Evaluation 94
4.3.3 Applications and Extensions of the Open-Source PCE 99
4.4 Multi-Layer Path Computation with PCE 100
4.4.1 Proposals for Multi-Layer Path Computation 101
4.4.2 Experimental Validation 105
4.5 Summary 112

5 Multi-Layer Network Orchestration 115
5.1 Supporting Publications 116
5.2 Network Management Challenges 117
5.2.1 The ONE Adapter Approach 118
5.3 Architecture of the ONE Adapter 119
5.3.1 Typical Execution of a Multi-layer Operation in the ONE Adapter 120
5.3.2 ONE Core Modules 121
5.3.3 ONE Auxiliary Modules 124
5.3.4 Implementation Features 129
5.3.5 Examples of Multi-layer Orchestration with the ONE Adapter 131
5.4 Summary 134

6 Conclusion 137

Bibliography 141

List of Symbols 153

Acronyms 159

1

Introduction

The last decade has witnessed the emergence of numerous paradigm-shifting Internet applications, including social media, cloud computing, big data, video streaming and mobile applications. The unprecedented growth of such applications has forced network operators to cope with ever-increasing demands for bandwidth and Quality of Service (QoS), which in turn has driven across-the-board innovations in networking. Innovations in optical transport have focused primarily on the development of cheap and high-bandwidth transmission systems (100 Gbps and beyond), flexible capacity utilization in the optical ([7]) as well as layer-2 technologies ([8, 9]) and control and management solutions for dynamic provisioning of services. However, almost all *user* services are currently provisioned and managed over IP/MPLS networks, and consequently, there is significant interest in developing solutions that can efficiently integrate and utilize the capabilities of IP/MPLS and optical transport infrastructure.

This emerging trend is clearly exhibited in numerous areas, including the latest service offerings of network operators, the next generation of integrated IP-optical network equipment as well as upcoming Software Defined Networking (SDN) based solutions. In terms of service offerings, operators have rolled out dynamic circuit services that can be exploited by users or applications to create dedicated circuits for transporting large volumes of traffic. The initial adoption and deployment of such services was seen in scientific research networks such as ESNet [10], Internet2 [11] and GEANT [12]. Dynamic circuit services are now also being offered by large vendors such as AT&T [13] and Verizon [14]. All of these services allow users to create on-demand end-to-end dynamic optical circuits for specific applications, while regular Internet traffic is routed over the best-effort IP infrastructure. Equipment vendors have also identified the need for IP-optical integration and some equipment vendors such as Juniper [15] and Cisco [16] are introducing integrated IP-optical devices aimed at lowering equipment costs and energy consumption while scaling beyond terabit capacity in a single chassis. Operators and equipment vendors alike have also identified the coordination of management operations in IP and optical transport networks as a major challenge, and are actively participating in the design and development of SDN standards such as [17, 18, 19, 20], which have the potential to support programmable management of multi-layer multi-vendor networks.

While dynamic circuit services are available for end-users, and the optical infrastructure is co-located with IP routers, network operators do not employ dynamic optical circuits for supporting operations in the IP network. Instead, *headroom practices* are commonly used in IP networks, where the capacity of IP links is designed to ensure that the utilization at peak loads on IP links is between 30% to 40% [21]. Such practices are clearly not cost-efficient, but ensure that service guarantees are met even in the case of unforeseen conditions such as multiple failures or external events leading to unusually large traffic volumes. However, with the expected exponential increase in Internet traffic, such practices will not be economically viable in the near future.

The introduction of dynamic optical circuits in IP networks has been studied [22] and even demonstrated in controlled environments, but their application in production networks presents numerous management challenges. Some of these challenges stem from the nature of IP routing itself: for example, a change in the IP network topology requires the IP routing protocols such as Open Shortest Path First (OSPF) to re-converge to a new routing configuration, which can take as long as 30 seconds [23] and can lead to the formation of temporary routing loops during the routing re-convergence process. Changes in IP routing also require the reconfiguration of associated management infrastructure such as measurement systems, Service Level Agreement (SLA) monitoring and alarm correlation functions [24], which is non-trivial. Some of the challenges involving the coordination of operations such as routing changes can possibly be addressed using SDN frameworks. However, given that the technology is still immature, it is unlikely that SDN will completely replace traditional network control and management paradigms in the near future.

As a result, it is critical to accurately model and quantify the interplay between the introduction of optical circuits (as new IP links) and the subsequent effects on IP routing in order to use dynamic optical circuits in IP networks. This implicitly requires *accurate modeling of the existing IP routing protocols* when evaluating the location and capacity of optical circuits to be employed in the IP network. Current research has primarily used Integer Linear Program (ILP) based models for evaluating the same, but have not modeled the features of the two technologies accurately, especially the characteristics of IP forwarding and routing protocols in the network. Basic features of IP networks, such as destination address based forwarding lookups have been ignored in the current models. The modeling of routing protocols such as OSPF [25] is also simplistic, only considering the requirements on using the shortest cost path, and ignoring constraints on the nature of routing protocol convergence. As a result, the application of these models to compute the location and capacity of optical circuits in IP networks was not accurate. Current solutions also do not take into consideration routing changes and the consequent management overhead associated with them, which poses a significant management overhead. This thesis presents an analytical framework that models fundamental features of IP routing and forwarding for existing standard schemes such as Explicit Routing (ER) and Shortest Path First Routing (SPF) and quantitatively demonstrates that the inaccurate modeling of IP routing can lead to significant discrepancies in the requirements on optical circuits in terms of optical capacity and placement. The framework also tackles systems challenges such as the requirement to decommission dynamic optical circuits and the need to compute solutions under unknown IP traffic matrix conditions that have not been addressed sufficiently in current research. A novel solution termed the Optical Bypass (BY) is presented which is designed to ensure that the introduction of optical circuits has very little effect on IP routing. The numerical study presented in this thesis demonstrates the Optical Bypass solution to be more efficient than SPF in terms of required optical capacity, number of affected routes as well as computational complexity, making it ideally suited for IP-Optical integration in dynamic network scenarios.

IP-optical integration also requires the coordination of operations across multiple network management systems, which is non-trivial. Management systems and practices employed for operating the IP and the optical network infrastructure differ to the extent that most carriers treat IP and optical transport networks as different administrative entities. This separation implies that any operation spanning the IP and optical networks requires manual interaction between human operators for both networks, which leads to high provisioning times and significant management overhead.

Given this situation, automation across multiple network layers is likely to be achieved by coordinating operations between entities in the different networks. Operators and vendors are also driving standardization activities for third-party systems that can perform specific management operations in different network layers. One such management subsystem is the Path Computation Element (PCE) which has emerged as the de-facto standard for constrained path computation. This thesis presents the design and implementation specification of the first open-source PCE, which is an essential tool for developing and testing system, protocol and architectural features associated with the PCE. The thesis also presents solutions that demonstrate multi-layer operations as a coordination between management entities in different network layers. The first solution demonstrates the ability of the PCEs in the IP and the optical network layers to interact with each other and compute a multi-layer path, and outlines some implementation and standardization challenges associated with the same. The second solution, named the ONE adapter, is a middleware architecture that facilitates programmable orchestration between network management (sub)systems in different network layers. The thesis presents an overview of the ONE adapter architecture and demonstrates the capability of the proposed architecture to facilitate specific aspects of IP-optical integration proposed in this thesis.

1.1 Thesis Contributions

The work in this thesis contributes to a number of modeling and systems aspects pertaining to integration of IP and dynamic optical transport networks in operational network settings. The primary contributions focus on accurately modeling the behavior of IP routing protocols that would give us fundamental understanding to system design, which has typically not been considered till now. The thesis also addresses some critical operational aspects of IP and optical networks which have been studied in isolation in the context of each network, but require a different approach in a multi-layer system. The thesis also presents novel architectures and implementation of components that facilitate multi-layer operations for IP-optical integration such as multi-layer path computation and provisioning. The solutions presented are designed for deployment in a commercial network operator's ecosystem, and therefore build upon existing standards and frameworks when possible. One of the proposed subsystems implementation is also open-source.

Specific contributions from the thesis can be divided into four major categories and are described next.

1.1.1 Modeling Effects of Dynamic Optical Circuit Provisioning on IP Routing

The ILP-based modeling presented in this thesis identifies numerous limiting assumptions made in current research, when modeling IP routing in conjunction with dynamic optical circuit provisioning, and critically evaluates the validity of these constraints, which are often taken for granted. For instance, the model presented in this thesis evaluates the differences in terms of forwarding capability of traditional IP and MPLS capable systems, and proposes new forwarding constraints that can be applied to these systems. The two models presented in this context, namely the Explicit Routing (ER) and Explicit Routing under Destination based Forwarding (ER-D) indicate the need to adapt the formulation to the capabilities of the underlying technology.

The model also identifies key aspects of SPF-based IP routing protocols, which have not been addressed before. Models for SPF routing under IP-optical integration approaches traditionally only consider the issue of shortest-path routing, and do not consider the way the routing protocol sets up routing rules, which constrains forwarding decisions to be made only on the basis of the destination address, or the events which can trigger a computation of a new route, which is only triggered when a *shorter* route is found in the network. The model presented in this thesis presents novel constraints to model these features and demonstrates numerically that existing models that do not incorporate constraints on the same compute solutions that are not accurate in real systems and generally under-estimate the requirements on optical capacity in the network.

The thesis also identifies some operational challenges pertaining to IP-optical integration that have not been incorporated in existing models. One such challenge involves identifying if a dynamic optical circuit should be switched-off, which has not been studied to date. The model presented in this thesis contains a new objective function definition that can evaluate if new dynamic circuits should be employed as well as if existing dynamic optical circuits should be decommissioned irrespective of the routing scheme used in the IP network. Another challenge involves the computation of solutions under unknown traffic matrix conditions. Most existing models assume the knowledge of IP traffic matrix in their analysis, but determining the traffic matrix in IP networks is non-trivial. The thesis proposes a novel formulation to compute the location and placement of dynamic optical circuits under unknown traffic matrix conditions using easily available traffic measurements.

1.1.2 Optical Bypass - A Novel Mechanism for Use of Dynamic Optical Circuits in IP Networks

The thesis proposes and presents the Optical Bypass (BY) scheme for introducing dynamic optical circuits in IP networks. While the term itself has been used in different contexts, the Optical Bypass (BY) proposal in this thesis contains distinct features that make it suitable for application in core IP networks. The mechanism is designed to minimize the impact of the introduction of dynamic optical circuits on IP routing, so as to be applicable in dynamic network scenarios. The thesis presents the concept and an ILP-based formulation for the BY mechanism and provides a detailed quantitative comparison with the traditional ER and SPF schemes, showing that the BY mechanism requires lower optical capacity with very little impact on IP routing and significantly lower time complexity as compared to SPF-based schemes in a number of network scenario. The proposed mechanism for computation under unknown traffic matrix conditions is also ideally suited for the Optical Bypass approach, and numerical results show that the model can compute near-optimal results for the Optical Bypass approach under unknown traffic matrix conditions.

1.1.3 Systems and Standards Challenges Associated with Coordinated Multi-layer Path Computation

The Path Computation Element (PCE) has emerged as the de-facto standard for constrained path computation across a wide spectrum of network technologies, and current standardization activities for the PCE encompass a plethora of network scenarios including multi-domain, multi-technology and multi-layer path computation. A major challenge, however, remains to evaluate the performance of the PCE in the different network scenarios. To this end, the *first open-source PCE*, which was developed as a part of this thesis, is presented. The thesis outlines the modular and extensible software architecture of the implementation, which allow operators to easily customize the PCE to different network technologies and also presents a performance study to demonstrate the scalability of the implementation.

The open-source PCE implementation is used to perform the first experimental evaluation of current PCE-based proposals for multi-layer path computation. The study identifies the candidate solution best suited for multi-layer path computation in the commercial network scenarios, and highlights standardization gaps and challenges with facilitating the same in real networks.

The capabilities of the PCE architecture and the proposed implementation are ideally suited for application in legacy network management architectures as well as upcoming SDN-based architectures. The ability to offload complex computation processes to a standardized third-party system significantly reduces the effort in developing SDN controllers, while the extensible nature of the PCE implementation also allows features such as topology description and update mechanisms to be integrated with mechanisms supported by upcoming SDN controllers. The use of PCE-based architectures also eases the transition from legacy to SDN-capable systems by ensuring that paths computed against service requests in an operator's ecosystem remain unchanged during the transition process.

1.1.4 Coordinated Orchestration of Management Operations Between IP and Transport NMSs

In the context of system design and implementation, the thesis outlines the differences in current management practices and NMSs used for managing IP and optical transport networks, which have resulted in administrative separation of these networks within carrier ecosystems. This thesis proposed the architecture of the ONE adapter, which is a middleware solution currently being developed in an EU project. The Service Oriented Architecture (SOA) for the ONE adapter proposes a framework for programmable orchestration between existing IP and optical transport NMSs as well as third-party management subsystems such as the PCE for performing multi-layer operations. This thesis presents the basic architecture of the ONE adapter and the detailed design of specific modules that were developed at TU Braunschweig as a part of the ONE project. The thesis also outlines application scenarios based on the Optical Bypass solution as well as the PCE-based multi-layer path computation solution that were proposed and demonstrated on actual network equipment. The use of Optical Bypass for offloading IP traffic has also been demonstrated in a real network as a joint demonstration between network operators and vendors [26].

1.2 Supporting Publications

1.2.1 Journal Articles

1. O. Gonzalez de Dios, V. Lopez, M. Cuaresma, F. Munoz, M. Chamania, A. Jukan,"Coordinated Computation and Setup of Multi-layer Paths via Inter-layer PCE Communication: Standards, Interoperability and Deployment," to appear in **IEEE Communications Magazine**, 2013 (Telecommunications Standards Series).

2. M. Chamania, A. Jukan, "A Comparative Analysis of the Effects of Dynamic Optical Circuit Provisioning on IP Routing," to appear in **IEEE/ACM Transactions on Networking**, 2013.

3. M. Caria, M. Chamania, A. Jukan, "A Comparative Performance Study of Load Adaptive Energy Saving Schemes for IP-over-WDM Networks" **IEEE/OSA Journal of Optical Communications and Networking**, vol.4, no. 3, pp. 152-164, 2012.

4. M. Chamania, M. Drogon, A. Jukan, "An Open-Source Path Computation Element (PCE) Emulator: Design, Implementation and Performance," **IEEE/OSA Journal of Lightwave Technology**, vol.30, no.4, pp. 414-426, 2012.

5. S. Greco Polito, S. Zaghloul, M. Chamania, A. Jukan, "Inter-Domain Path Provisioning with Security Features: Architecture and Signaling Performance," **IEEE Transactions on Network and Service Management**, vol.8, no.3, pp.219-233, 2011.

6. M. Chamania, M. Caria, A. Jukan, "Achieving IP Routing Stability with Optical Bypass," **Optical Switching and Networking**, vol. 7, no. 4, pp. 173-184, 2010.

7. M. Chamania, X. Chen, A. Jukan, F. Rambach, M. Hoffmann, "An adaptive inter-domain PCE framework to improve resource utilization and reduce inter-domain signaling," **Optical Switching and Networking,** vol.6, no.4, pp.259-267, 2009.

8. M. Chamania, A. Jukan, "A Survey of Inter-Domain Peering and Provisioning Solutions for the Next Generation Optical Networks," **IEEE Communications Surveys and Tutorials**, vol.11, no.1, pp.33-51, 2009.

1.2.2 Conferences and Workshops

1. S. Martinez, V. Lopez, M. Chamania, O. Gonzalez de Dios, A. Jukan, J. P. Fernandez-Palacios, "Assessing the Performance of Multi-Layer Path Computation Algorithms for different PCE Architectures," **OSA Optical Fiber Communication Conference (OFC) / National Fiber Optic Engineers Conference (NFOEC)**, 2013.

2. M. Drogon, M. Chamania, A. Jukan, M. Yannuzzi, X. Masip-Bruin, V. Lopez, O. Gonzalez, M. Maciejewski, M. Roth, C. Brunn, J. Altmann, "Towards Automated Interactions between the Internet and the Carrier-Grade Management Ecosystems," **Future Network and Mobile Summit**, 2013.

3. X. Chen, M. Chamania, A. Jukan, "On the Effectiveness of Optical Parallel Transmission in IP Offloading," **Networks and Optical Communications (NOC)**, 2012.

4. M. Chamania, O. Gonzalez de Dios, V. Lopez, M. Cuaresma, M. Drogon, A. Jukan, X. Masip-Bruin, M. Yannuzzi, "Coor-

dinated Computation of Multi-layer Paths via Inter-layer PCE Communication: Standards, Interoperability and Deployment," **IEEE International Conference on Communications (ICC)**, 2012.

5. M. Chamania, E. Demirbilek, A. Jukan, X. Masip-Bruin, M. Yannuzzi, "Using BPEL Workflow Processing for Cross-Layer Orchestrations in IP-over-Optical Networks: A Proof of Concept," **IEEE/IFIP Network Operations and Management Symposium (NOMS)**, 2012.

6. M. Yannuzzi, A. Jukan, X Masip-Bruin, M. Chamania, R. Serral-Gracia, Victor Lopez, O Gonzalez de Dios, M. Maciejewski, J. Altmann, M. Roth, C. Brunn, A. Azanon,"The Internet and Transport Network Management Ecosystems: A Roadmap Towards Convergence," **Optical Network Design and Modeling (ONDM)**, 2012. (*Invited paper*)

7. M. Chamania, M. Caria, A. Jukan, "A Comparative Performance Analysis of IP Traffic Offloading Schemes over Dynamic Circuits," **IEEE INFOCOM**, 2011.

8. **POST-DEADLINE PAPER** M. Chamania, M. Drogon, A. Jukan, "Lessons Learned From Implementing a Path Computation Element (PCE) Emulator," **OSA Optical Fiber Communication Conference (OFC) / National Fiber Optic Engineers Conference (NFOEC)**, 2011.

9. X. Chen, M. Chamania, A. Jukan, "A Backward-Compatible Inter-domain Multipath Routing Framework," **IEEE Conference on Computer Communications Workshops (INFOCOM Workshops)**, 2011.

10. M. Caria, M. Chamania, A. Jukan, "To Switch On or Off: A Simple Case Study on Energy Efficiency in IP-over-WDM Net-

works," **IEEE High Performance Switching and Routing (HPSR)**, 2011.

11. M. Caria, M. Chamania, A. Jukan, "Trading IP Routing Stability for Energy Efficiency: A Case for Traffic Offloading with Optical Bypass," **Optical Network Design and Modeling (ONDM)**, 2011.

12. M. Chamania, M. Caria, A. Jukan, "Effective Usage of Dynamic Circuits for IP Routing," **IEEE International Conference on Communications (ICC)**, 2010.

13. S. Greco-Polito, D. Gebbers, M. Chamania, A. Jukan, "A New NSIS Application for LSP Setup with Security Features," **IEEE International Conference on Communications (ICC)**, 2010.

14. O. Tipmongkolsilp, M. Chamania, A. Jukan, "Network-Centric Media Composition under Extreme Bandwidth Requirements for High Quality Video Streaming," **ACM Supercomputing (SC)**, 2010.

15. M. Chamania, M. Caria, A. Jukan, "Achieving IP Routing Stability with Optical Bypass," **IEEE Advanced Networks and Telecommunication Systems (ANTS)**, 2009.

16. M. Chamania, X. Chen, A. Jukan, F. Rambach, C. Gruber, M. Hoffmann, "Embedding Optical Ethernet Services within the Path Computation Element Framework: The 100GET Approach," **OSA Optical Fiber Communication Conference (OFC) / National Fiber Optic Engineers Conference (NFOEC)**, 2009.

17. S. Greco-Polito, M. Chamania, A. Jukan, "Extending the PCE Framework for Authentication and Authorization in Multi-

domain GMPLS Networks," **IEEE International Conference on Communications (ICC)**, 2009.

18. A. Gumaste, M. Chamania, A. Jukan, "An End-to-End Addressing and Routing Scheme in all-Ethernet Wide-Area Networks," **IEEE International Conference on Communications (ICC)**, 2009.

19. X. Chen, M. Chamania, A. Jukan, A. Drummond, N. L. S. Fonseca, "On the Benefits of Multipath Routing for Distributed Data-intensive Applications with High Bandwidth Requirements and Multidomain Reach," **Communication Networks and Services Research Conference (CNSR)**, 2009.

20. X. Chen, M. Chamania, A. Jukan, A. C. Drummond, N. L. S. da, Fonseca, "QoS-Constrained Multi-path Routing for High-End Network Applications," **IEEE High-Speed Networks Workshop**, 2009.

21. T. Michaelis, M. Duelli, M. Chamania, B. Lichtinger, F. Rambach, S. Turk, "Network Planning, Control and Management Perspectives on Dynamic Networking," **European Conference on Optical Communication (ECOC)**, 2009. (*Invited paper*)

22. M. Chamania, X. Chen, A. Jukan, F. Rambach, C. Gruber, M. Hoffmann, "Adaptive Advance Reservation Based Inter-Domain Framework," **IEEE Advanced Networks and Telecommunication Systems (ANTS)**, 2008.

1.3 Thesis Organization

This thesis is structured in six chapters. After the introduction, Chapter 2 presents an overview of the existing techniques for introducing dynamic optical circuits in IP networks and presents an ILP-based model that addresses the shortcomings of the same. Chapter 3 introduces and presents the modeling for the Optical Bypass model, followed by a numerical analysis comparing the performance of the Optical Bypass model with traditional schemes presented in Chapter 2. Chapter 4 presents an overview of the open-source PCE, and goes on to demonstrate the application of the same for implementing multi-layer path computation and provisioning. Chapter 5 presents the architecture of the ONE adapter for facilitating programmable coordination across multiple network layers. This chapter also highlights the capability of the ONE adapter to enable features such as dynamic IP offloading with Optical Bypass and multi-layer PCE based service provisioning presented in Chapters 3 and 4 respectively. Finally, Chapter 6 concludes the thesis and provides directions for further work.

2

Modeling IP-Optical Integration Under Traditional IP Routing

The concept of dynamically introducing optical circuits in IP networks has been proposed and studied for over a decade, but has not been employed in operator networks. Current research has proposed models to determine the location and capacity of optical circuits that should be used to address overloading in IP networks, but assumptions made in the proposed model, especially on the behavior of IP routing differ significantly from real application scenarios.

The inaccurate modeling of IP routing not only introduce inconsistencies in the computed and the required optical capacity and placement, but also present a significant management challenge for network operators. For example, in a network employing OSPF, the introduction of a new IP link leads to routing re-convergence, which can take as long as 30 seconds and can lead to creation of temporary routing-loops [23]. Changes in routing also lead to significant changes in network management configuration: for example, alarm correlation functions to identify the location and nature of a fault need to be re-computed with every routing change in the network [24]. It is therefore essential to model IP routing mechanisms as accurately as possible, and understand the inter-play between optical capacity demands and routing re-configurations exhibited by different mechanisms available for introducing optical circuits in IP networks.

This chapter presents ILP-based modeling for introducing optical circuits in IP networks under the two traditional IP routing schemes, namely Explicit Routing (ER) and Shortest Path First Routing (SPF) mechanisms, which have been presented in [1]. The chapter highlights the limitations of existing research in modeling IP routing, and presents novel constraints for incorporating features such as destination-based forwarding and routing protocol reconvergence. These features are intrinsic to the behavior of IP forwarding and routing protocols, but have not been studied in existing research on IP-optical integration. The proposed model also presents a new objective function that can compute decommissioning of established dynamic optical circuits along with the introduction of new circuits. Finally, a novel formulation is proposed for computation under unknown traffic matrix conditions, which uses known traffic measurements to compute solutions with deterministic upper bounds on required optical capacity.

2.1 Supporting Publications

1. M. Chamania, A. Jukan, "A Comparative Analysis of the Effects of Dynamic Optical Circuit Provisioning on IP Routing," to appear in **IEEE/ACM Transactions on Networking**, 2013.

2. M. Chamania, M. Caria, A. Jukan, "A Comparative Performance Analysis of IP Traffic Offloading Schemes over Dynamic Circuits," **IEEE INFOCOM**, 2011.

2.2 Limitations of Existing IP-Optical Modeling Solutions

The modeling of IP-Optical integration has been studied primarily as an optimization problem [27, 28, 29, 30]. In typical formulations, IP networks and optical transport networks are treated as two separate entities, and a subset of IP routers are co-located with optical transport switches. It is assumed that an inter-connection can be established between IP routers by creating a circuit between the co-located optical switches in the transport network, and this interconnection is either advertised as a new IP link, or is used to boost the capacity of an existing IP link. The capacity of optical circuits is limited to a small subset of available circuit capacities in the network. These formulations can also be extended to include technology-specific constraints such as wavelength continuity [31] and OSNR bounds [32] for routing in WDM networks. Recent years have also witnessed the emergence of new transport technologies such as carrier Ethernet [9] and flexi-grid optical networks [7], which allow flexible capacity allocation, and can also make use of mechanisms such as multi-path routing [33], [34].

Routing of traffic in the IP network in these models makes assumptions that are not accurate in real IP networks. The models assume that a route from a source (ingress) router to a destination (egress) router uses a single (unique) path, and do not otherwise constrain forwarding decisions for IP traffic. The models also assume complete knowledge of the IP traffic matrix i.e. the traffic between all possible (ingress, egress) router pairs in the network.

These formulations do not take into consideration many of the intrinsic features of IP routing and forwarding, and consequently cannot accurately compute the outcome of introducing optical circuits in IP networks. The three major features of IP routing and forwarding not considered in the existing formulations include *destination-based forwarding*, *shortest path routing* and constraints on *routing* *re-convergence*. Out of these, the shortest path routing constraint has been applied individually in some models, but the modeling of IP routing, especially in the presence of routing protocols such as OSPF [25] must incorporate all of these constraints in order to accurately evaluate the behavior of IP routing. As demonstrated later in Section 3.5, the difference in terms of required optical capacity can be significant if even one of the three constraints is not considered when modeling the behavior of IP routing.

The existing network models also present formulations that always assume that a new optical circuit may be added to the network, but do not incorporate the process of *decommissioning dynamic optical circuits*, which is a critical requirement for using dynamic optical circuits in IP networks.

Most of the existing research also assumes that the complete IP traffic matrix [1] is known, which is impractical in large networks due to the high costs and monitoring overhead associated with the same [35]. Traditional mechanisms to overcome this problem involve the estimation of IP traffic matrices which are then substituted into the original formulation. However, inaccurate estimation of the IP traffic matrices could lead to computation of solutions which do not resolve the overloading of links. Similar conclusions were presented in the area of optimal traffic engineering [36] and optical bypasses [37] indicating that traffic estimates are not accurate enough to be used as a viable substitute for the actual traffic matrix. Some proposals such as [38] propose to introduce optical circuits to force routing changes in the network, which in turn provides more measurements and improves the estimation of the traffic matrix. However, this practice is not suitable in carrier networks due to the high management overhead associated with network reconfiguration and the possible loss of traffic due to the formation of temporary routing

[1] The IP traffic matrix is defined using actual traffic value between *all* ingress-egress router pairs, i.e. traffic entering the IP network at an ingress router, and leaving the network at the egress router.

loops during routing protocol re-convergence.

The rest of this chapter presents an ILP-based model that addresses the shortcomings of current research. The presented ILP model incorporates the capability to introduce and *switch-off* dynamic optical circuits in its objective function, and presents constraints for modeling destination-based forwarding, shortest path routing and routing re-convergence. The chapter also presents a formulation that can use easily available traffic measurements in IP networks, such as IP link loads (using the Simple Network Management Protocol (SNMP) [39]), and traffic in Virtual Output Queues (VOQ) on IP routers[2] to compute the requirement on optical capacity in the absence of traffic matrix information.

2.3 Network Model Description

In this section, we first describe the generic IP-over-optical network scenario and use it to motivate and justify the assumptions made on the mathematical network model.

2.3.1 Network Model Assumptions

The generic IP-over-optical network scenario is presented in Fig. 2.1. Here, routers in the IP network are co-located with optical switches, and IP links are provisioned using fixed optical circuits or leased lines. Traffic demand is served over the IP network, and additional optical capacity, if required, is installed to meet the traffic demand. Our application scenario is a core network, where every router acts as the edge router to other transit/stub networks and thus traffic is present between all pairs of routers.

[2] Access to measurements from Virtual Output Queues is typically vendor-specific, and different techniques, such as [40] can be applied in order to extract these measurements from devices.

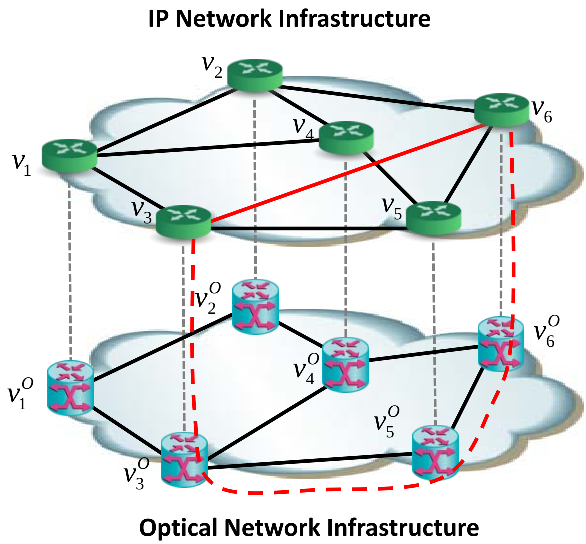


Figure 2.1: Generic IP-over-Optical Network Scenario used to model the network

For the simplicity of presentation, we assume that every router in the IP network is connected to a unique optical switch in the optical network: for example, as shown in Fig. 2.1 the router v_i is always connected to the corresponding switch v_i^O, and we assume that the indexes of interconnected routers and switches in both the layers are the same [3]. There is, however, no constraints on either networks to have the same number of nodes/links/inter-connectivity.

The fixed optical circuits (leased lines) are assumed to be static, and in order to add additional capacity, dynamic optical circuits are

[3]Note that the same indexes are used to simplify the presentation of the computation model, as in this case an optical circuit between v_x^O to v_y^O can be easily mapped to an IP link between v_x and v_y

introduced in the IP network. Traditional transport networks such as Optical Transport Networks (OTN)[8] and Synchronous Digital Hierarchy (SDH)[41] support only a small number of services with fixed circuit capacity, and in our model we assume the same. We also support the aggregation of multiple optical circuits, which can be seen in transport network technologies such as flexi-grid [7] and virtual concatenation [42].

The model is formulated as an Integer Linear Program (ILP), and attempts to compute the requirement on optical capacity to alleviate "overloading" in the network. In the model, the IP network is presumed to be overloaded when the capacity utilization of one or more IP links is higher than a fixed threshold (α), and the model attempts to compute the location of new IP link(s), their capacity and the associated routing changes (based on the IP routing mechanism employed) to eliminate overloading in the IP network, while optimizing cost based on the resources required in the optical network.

The model assumes that IP links are always bi-directional. This is motivated primarily by the fact that many functions such as OSPF [25] neighbor discovery and Bi-directional Forwarding Detection [43] require bi-directional connectivity in the IP network. However, directional optical circuits are used to provision the IP link capacity, which implies that IP links can have asymmetric upstream and downstream capacity. Routes in the IP network always use a *single path* from a source router to a destination router, which are either configured explicitly or via routing protocols such as OSPF. The actual forwarding capabilities available on IP routers can also determine routing behavior: for example, in an network that supports Multiprotocol Label Switching (MPLS) it is possible to explicitly configure MPLS tunnels between all source-destination router pairs, providing full routing flexibility, while in a traditional IP network, static routing rules only allow the forwarding decisions to be made based on the destination address[4]. The formulation presented takes into account these features and presents different formulations for IP routing that can adapt to the capabilities of the available technologies.

When modeling SPF protocols, the model assumes that the link weight of a link (if it exists) is a fixed (known) constant. The model also assumes the use of single-path routing under the SPF routing protocol. Equal Cost Multi-Path routing (ECMP) flavors are supported by most vendors under OSPF routing, but are typically not employed in core networks because of the inconsistency in how different vendors manage distribution of traffic across multiple paths, especially in the presence of non-IP MPLS traffic (e.g. PseudoWire service).

The formulation presented provides constraints for computation under known traffic matrix conditions, and provides constraints for using traffic measurements for enforcing IP link capacity constraints under unknown traffic matrix conditions.

2.3.2 Network Architecture Formulation and Constraints

The model assumes that the topology $\mathcal{G}(\mathcal{V}, \mathcal{E})$ in the IP network consists of a set of routers $\mathcal{V}$ with unique routers $v_i \in \mathcal{V}$. As specified before, fixed links in the IP network are assumed to be static i.e. these links cannot be established or torn-down, while dynamic optical circuits can be introduced/removed from the network. The set $\mathcal{E}$ consists of all the fixed IP links in the IP network, with $e_{ij} \in \mathcal{E}$ indicating an IP link from v_i to v_j.

In the model, dynamic optical circuits established in the optical transport network can be used to create links in the IP network. We

[4] While mechanisms such as Policy-based routing [44] can be used to forward IP traffic based on both source and destination address, they are typically not used inside the core IP network due to the additional overhead associated with the look-up of source IP addresses during forwarding.

assume that dynamic circuit services available only support a fixed (small) number of discrete circuit granularities given by the set $\mathcal{T}$. Each unique circuit type $t \in \mathcal{T}$ is associated with a circuit capacity C^O_t which is used to compute the capacity of the associated IP link. The model also allows the aggregation of one or more (maximum N) optical circuits to create a single IP link.

A boolean variable X_{xy} $(x \neq y)$ indicates if one or more dynamic optical circuits have been established from v_x to v_y. The circuit(s) can have different granularities, and a positive integer variable X^t_{xy} $(x \neq y)$ indicates the number of circuits of type $t \in \mathcal{T}$ that are established from v_x to v_y. The relationship between X_{xy} and X^t_{xy} is defined using the following equations:

$$\forall v_x, v_y \in \mathcal{V}, x \neq y \ : \ \sum_t X^t_{xy} \leq N \tag{2.1}$$

$$\forall v_x, v_y \in \mathcal{V}, x \neq y, t \in \mathcal{T} \ : \ X_{xy} \leq \sum_t X^t_{xy} \tag{2.2}$$

$$\forall v_x, v_y \in \mathcal{V}, x \neq y, t \in \mathcal{T} \ : \ X_{xy} \geq \frac{\sum_t X^t_{xy}}{N} \tag{2.3}$$

Eq. (2.1) constraints the total number of circuits of any granularity that are established from v_x to v_y to be less than N . Eqns. (2.2) and (2.3) constrain the relationship between X_{xy} and X^t_{xy} . Here, (2.2) constraints $X_{xy} = 0$ if no dynamic circuits are established from v_x to v_y, while (2.3) constrains $X_{xy} = 1$ if one or more dynamic circuits are established from v_x to v_y.

The formulation assumes that only one IP link can exist between a pair of routers v_x and v_y, and therefore optical circuits are used to either boost the capacity of an existing link or are used to create new IP links between previously non-neighboring routers. IP links created using fixed optical circuits are indicated using the constant $\hat{L}_{ij}$ $(i \neq j)$, with $\hat{L}_{ij} = 1$ indicating an IP link with fixed optical circuit from v_i to v_j. The boolean variable L_{ij} $(i \neq j)$ indicates if an IP link exists from v_i to v_j after the establishment of dynamic optical circuits using the following constraints:

$$\forall v_i, v_j \in \mathcal{V},\ i \neq j : \qquad L_{ij} \geq \hat{L}_{ij} \tag{2.4}$$

$$\forall v_i, v_j \in \mathcal{V},\ i \neq j : \qquad L_{ij} \geq X_{ij} \tag{2.5}$$

$$\forall v_i, v_j \in \mathcal{V},\ i \neq j : \qquad L_{ij} \leq \hat{L}_{ij} + X_{ij} \tag{2.6}$$

$$\forall v_i, v_j \in \mathcal{V},\ i \neq j : \qquad L_{ij} = L_{ji} \tag{2.7}$$

Eqn. (2.4) and (2.5) ensure that $L_{ij} = 1$ if either fixed ($\hat{L}_{ij} = 1$) or dynamic optical circuits ($X_{ij} = 1$) are used from v_i to v_j, while (2.6) ensures that $L_{ij} = 0$ if neither of the fixed or dynamic optical circuits exist between a pair of routers. Finally, constraint (2.7) ensures that all established IP links are bi-directional, i.e. if a link exists from v_i to v_j then a link must also exist from v_j to v_i.

The capacity of the IP link is determined by the capacity of the fixed and dynamic optical circuits between the two routers. The capacity of fixed optical circuits from v_i to v_j is given by the constant $\hat{C}_{ij}$ ($i \neq j$), which is 0 if no fixed optical circuits exist between the specified pair of routers. C_{ij} ($i \neq j$) is a positive variable indicating the total directional[5] capacity of an IP link from v_i to v_j, and is defined using (2.8). Based on the constraints, C_{ij} is given as the sum of the capacities of the fixed and the dynamic optical circuits.

$$\forall v_i, v_j \ \in \ V \ : \ C_{ij} = \hat{C}_{ij} + \sum_t X_{ij}^t \cdot C_t^O \tag{2.8}$$

2.3.3 Optimization Objective

The presented formulation attempts to minimize the cost associated with installing additional optical circuits in the IP network. In the initial discussion, we will assume that the cost of installing a single

[5] As specified before, while IP links are bi-directional, the upstream/downstream capacities are not required to be symmetric and are therefore computed separately in both directions.

optical circuit of type $t \in \mathcal{T}$ from v_x to v_y is given by a known constant $Cost^t_{xy}$. The objective function (2.9) minimizes the total cost of introducing optical circuits in the IP network when we assume that the system only contains fixed optical circuits (no dynamic optical circuits in use at the time of computation).

$$Min: \sum_t \sum_{xy} X^t_{xy} \cdot Cost^t_{xy} \tag{2.9}$$

Modified Objective for Decommissioning Dynamic Optical Circuits

In a network deploying *dynamic* circuits, the model needs to consider scenarios where dynamic optical circuits may already be in use in the network and may need to be decommissioned or switched-off. In this scenario, the objective function is modified, based on the following assumptions on the cost of dynamic optical circuits:

1. Re-use of existing (already setup) dynamic circuits incurs no additional cost
2. Introduction of new optical circuits is associated with a fixed cost $Cost^t_{xy}$
3. Decommissioning of an optical circuit is associated with a profit SW^t_{xy} .

Information about already installed dynamic optical circuits is given by $\hat{X}_{xy}$ and $\hat{X}^t_{xy}$ $(x \neq y)$, with $\hat{X}_{xy} = 1$ indicating that one or more dynamic optical circuits were installed from v_x to v_y, and the exact number and granularity of these circuits is given by $\hat{X}^t_{xy}$[6].

In case no dynamic optical circuits exist before the optimization $(\hat{X}_{xy} = 0)$, an optical circuit cannot be switched-off, and hence a contribution to the optimization can only come in the form of the

[6]Note that $\hat{X}_{xy}$ and $\hat{X}^t_{xy}$ only indicate *dynamic* optical circuits that have already been installed in the network before the model computes a new solution, and are different from *fixed* optical circuits.

cost of a new optical circuit. However, when one or more dynamic optical circuits were present before the optimization ($\hat{X}_{xy} = 1$), the formulation must evaluate if some of the circuits in the initial configuration were switched-off and also if new dynamic optical circuits were installed.[7]

In the formulation, the expression ($\hat{X}^t_{xy}$ - X^t_{xy}) is used to determine if dynamic optical circuits with granularity $t \in \mathcal{T}$ were installed or switched-off from v_x to v_y. For a specific circuit granularity, if existing circuits are switched off, $X^t_{xy} < \hat{X}^t_{xy}$ while in case new dynamic optical circuits are added, $X^t_{xy} > \hat{X}^t_{xy}$. A boolean variable Y^t_{xy} ($x \neq y$) is used to identify if the final configuration for a specific granularity $t \in \mathcal{T}$ indicates the installation of new circuits or the switch-off of existing circuits from v_x to v_y, with $Y^t_{xy} = 1$ if additional circuits are installed and $Y^t_{xy} = 0$ otherwise. Using the fact that the maximum number of dynamic optical circuits between a pair of routers, and consequently X^t_{xy} and $\hat{X}^t_{xy}$, are bounded by [0, N], (2.10) and (2.11) are used to define constraints on Y^t_{xy} . In the case that $X^t_{xy} > \hat{X}^t_{xy}$, (2.10) ensures that $Y^t_{xy} = 1$, while (2.11) ensures that $Y^t_{xy} = 0$ otherwise.

$$\forall v_x, v_y \in V, x \neq y \ : \ Y^t_{xy} \geq N^{-1} \cdot (X^t_{xy} - \hat{X}^t_{xy}) \tag{2.10}$$

$$\forall v_x, v_y \in V, x \neq y \ : \ Y^t_{xy} \leq 1 + N^{-1} \cdot (X^t_{xy} - \hat{X}^t_{xy}) \tag{2.11}$$

Based on the different cases presented above, the ILP objective function (2.12) considers three distinct cases, which are represented as the three distinct terms in the objective function. The first term considers the scenario where no dynamic optical circuits exist initially ($\hat{X}_{xy} = 0$), and computes the cost of adding new optical circuits using the expression $X^t_{xy} \cdot Cost^t_{xy}$ which is the same as the

[7] The assumption to aggregate multiple optical circuits of different granularities means that it might be possible to switch-off some circuits of a certain granularity $t \in T$ while installing new dynamic optical circuits of granularity $t' \in T$ ($t \neq t'$) between a given set of routers.

cost function used in (2.9). In case dynamic optical circuits exist before the optimization, ($\hat{X}_{xy} = 1$), the second term considers the case when new optical circuits are introduced for a particular granularity ($Y^t_{xy} = 1$). In this case, the number of new optical circuits introduced is given by ($X^t_{xy} - \hat{X}^t_{xy}$), and the associated cost for each new circuit given by $Cost^t_{xy}$. The third term is included to compute the switch-off profit, with $Y^t_{xy} = 0$ indicating that circuits may have been switched off. Here, ($\hat{X}^t_{xy} - X^t_{xy}$) gives the number of circuits of a particular granularity that were switched-off and SW^t_{xy} is the profit associated with switching off a single circuit for the specified granularity from v_x to v_y.

$$
\begin{aligned}
Min: \sum_{xy}\sum_{t}(1-\hat{X}_{xy}) \cdot X^t_{xy} \cdot Cost^t_{xy} \\
+\sum_{xy}\sum_{t}\hat{X}_{xy} \cdot Y^t_{xy} \cdot (X^t_{xy} - \hat{X}^t_{xy}) \cdot Cost^t_{xy} \\
-\sum_{xy}\sum_{t}\hat{X}_{xy} \cdot (1-Y^t_{xy}) \cdot (\hat{X}^t_{xy} - X^t_{xy}) \cdot SW^t_{xy} \quad (2.12)
\end{aligned}
$$

The objective function in (2.12) contains non-linear terms with the product $Y^t_{xy} \cdot X^t_{xy}$, which is linearized by introducing a new variable Z^t_{xy} ($x \neq y$, $0 \leq Z^t_{xy} \leq N$). (2.13), (2.14) and (2.15) are used to constrain Z^t_{xy} . Given that Y^t_{xy} is boolean, (2.13) constrains Z^t_{xy} to be less than N if $Y^t_{xy} = 1$ and constrains Z^t_{xy} to be 0 otherwise, while (2.14) ensures that Z^t_{xy} is bounded by X^t_{xy}. Using these two constraints, (2.15) ensures that $Z^t_{xy} = X^t_{xy}$ if $Y^t_{xy} = 1$ and is equal to 0 otherwise.

$$\forall v_x, v_y \in V, t \in T \;:\; Z^t_{xy} \leq N \cdot Y^t_{xy} \quad (2.13)$$

$$\forall v_x, v_y \in V, t \in T \;:\; Z^t_{xy} \leq X^t_{xy} \quad (2.14)$$

$$\forall v_x, v_y \in V, t \in T \;:\; Z^t_{xy} \geq X^t_{xy} - (1 - Y^t_{xy}) \cdot N \quad (2.15)$$

2.4 Routing in IP Networks

When dynamic optical circuits are introduced as new IP links in the network, routing of traffic in the IP network can be changed either manually or via the routing protocol operating in the network. This section presents constraints for two different classes of routing mechanisms, namely Explicit Routing (ER) and Shortest Path First Routing (SPF). Both formulations are developed based on the assumption of a single unique route between a source and a destination router as stated in Section 2.3.1. Based on this assumption, a boolean variable r_{ij}^{sd} ($s \neq d$, $i \neq j$) is used to indicate if the route from v_s to v_d uses a link from v_i to v_j. As new links in the IP network can be established in the IP network using dynamic optical circuits, the variable r_{ij}^{sd} is constrained by (2.16) to ensure that a route can only choose a link from v_i to v_j if $L_{ij} = 1$ which indicates that an IP link exists between v_i and v_j.

$$\forall v_s, v_d, v_i, v_j \in \mathcal{V} : r_{ij}^{sd} \leq L_{ij} \tag{2.16}$$

As specified in Section 2.3.1, the model constrains the link utilization for all links in the IP network to be less than α . In case the traffic matrix is known, the traffic from v_s to v_d is given by $\hat{\lambda}_{sd}$ and the traffic on any given link can be computed using the traffic value and the routing variable r_{ij}^{sd} . Using these parameters, the link utilization threshold can be expressed using (2.17), where the capacity of the link C_{ij} is defined in (2.8). The formulation in case the traffic matrix is not known is presented in Section 2.6.

$$\forall v_i,\ v_j \in \mathcal{V},\ i \neq j\ :\ \sum_{sd} r_{ij}^{sd} \cdot \hat{\lambda}_{sd} \leq \alpha \cdot C_{ij} \tag{2.17}$$

The ER mechanism presents formulation where routes can be configured manually (or via an external control program), while the SPF mechanism presents the formulation for describing the behavior of typical IP routing protocols.

2.4.1 Explicit Routing (ER)

As the name suggests, Explicit Routing allows the explicit configuration of a route from a source to the destination, as long as a unique route is available for all source-destination router pairs. Furthermore, the route should not contain any routing loops as this is not desirable in the network. The constraints (2.18), (2.19) and (2.20) ensure that a single route exists between every source and destination, and are also illustrated using Fig. 2.2.

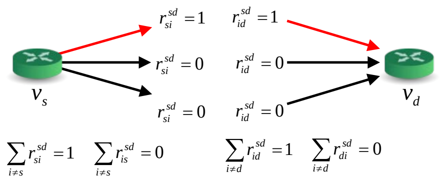
$$\forall v_s, v_d \in \mathcal{V},\ s \neq d\ :$$

$$\sum_{j \neq s} r_{sj}^{sd} = \sum_{j \neq d} r_{jd}^{sd} \ = \ 1 \tag{2.18}$$

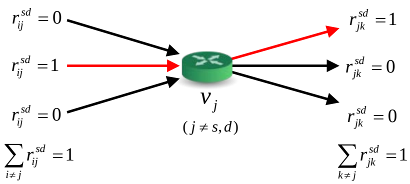
$$\sum_{j \neq s} r_{js}^{sd} = \sum_{j \neq d} r_{dj}^{sd} \ = \ 0 \tag{2.19}$$

$$\forall v_j \in \mathcal{V},\ j \neq s, d\ :\ \sum_{i \neq j} r_{ij}^{sd} = \sum_{k \neq j} r_{jk}^{sd} \leq 1 \tag{2.20}$$

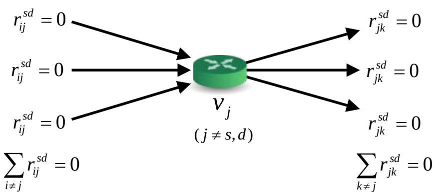
As seen in Fig. 2.2(a), for every source v_s and destination v_d ($s \neq d$), there must be exactly one outgoing link from v_s and one incoming link at v_d that are used for a route between the two, which is constrained using (2.18). At the same time, a source should not have any *incoming* link carrying the traffic from the said source, and a destination should not have any *outgoing* link that carries the traffic to the destination as defined in (2.19). Routing continuity at intermediate routers is presented using Figs. 2.2(b) and 2.2(c): In case an intermediate router lies on the route from v_s to v_d (Fig. 2.2(b)), there must be exactly one incoming link and one outgoing link that are used to route the traffic from v_s to v_d, while if the router does not lie on the routing path (Fig. 2.2(c)), no incoming or outgoing links from this router should be used to carry traffic from v_s to v_d. These conditions are enforced in the network using (2.20) which ensures that the number of incoming links carrying traffic for



(a) Routing Continuity at Source and Destination routers



(b) Routing Continuity at Intermediate routers when a route passes through them



(c) Routing Continuity at Intermediate routers when a route does not pass through them

Figure 2.2: Routing Continuity in IP networks

a particular source-destination pair at any arbitrary router v_j ($j \neq s, d$) should be equal to the number of outgoing links carrying traffic for the same source-destination pair, and the maximum number of such links should be restricted to 1. By restricting the maximum number of such links to 1, the constraint ensures that no routing loops are established in the network.

The routing continuity constraints presented in this section ensure the existence of a single unique route from v_s to v_d, and *forwarding* decisions at intermediate routers are made based on both the source and destination of traffic. Solutions computed under these assumptions can only be employed in networks using explicitly configured MPLS tunnels between all source-destination router pairs.

2.4.2 Explicit Routing Under Destination-based Forwarding (ED-D)

In traditional IP networks, as stated in Section 2.3.1, forwarding decisions for traffic at a router are based solely on the destination of the traffic. To ensure the same, a boolean variable $FT_i^d(j)$ ($i \neq d,\ j$)is used to model a forwarding table of a traditional IP router, with $FT_i^d(j) = 1$ indicating that all traffic at router v_i to destination v_d uses the link from v_i to v_j. The constraints in (2.21), (2.22) and (2.23) define the constraints on $FT_i^d(j)$ and its relationship with r_{ij}^{sd}

.

$$\forall v_d, v_i, v_j \in \mathcal{V},\ i \neq d : FT_i^d(j) \leq L_{ij} \tag{2.21}$$

$$\forall v_d, v_i \in \mathcal{V}, i \neq d : \sum_{j \neq i} FT_i^d(j) = 1 \tag{2.22}$$

$$\forall v_s, v_d, v_i, v_j \in \mathcal{V},\ i \neq d : FT_i^d(j) \geq r_{ij}^{sd} \tag{2.23}$$

(2.21) ensures that a forwarding decision at v_i can only use the link from v_i to v_j if it exists ($L_{ij} = 1$), while (2.22) ensures that exactly one outgoing link from v_i is chosen to route traffic to v_d ($i \neq d$).

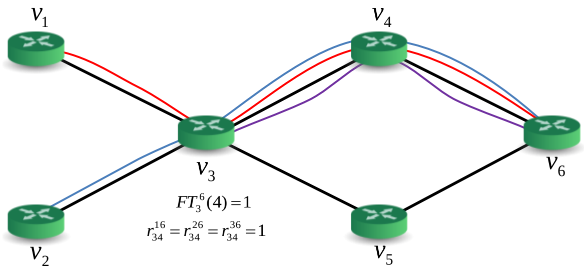


Figure 2.3: Example depicting destination-based forwarding in IP networks

Finally, (2.23) ensures that a route from v_s to v_d can only use the link from v_i to v_j if the forwarding table entry $FT_i^d(j) = 1$. The use of these constraints along with the routing continuity constraints (2.18), (2.19) and (2.20) ensures that the forwarding decision at any router can be defined solely based on the destination address.

An example describing the same can be seen in Fig. 2.3. In the figure, traffic to v_6 at v_3 can choose from one of two outgoing links to v_4 and v_5. In the case when destination-based forwarding constraints are not enforced, a valid solution may contain traffic from v_1 to v_6 using the route $v_1 \rightarrow v_3 \rightarrow v_4 \rightarrow v_6$, while the traffic from v_2 to v_6 may use the route $v_2 \rightarrow v_3 \rightarrow v_5 \rightarrow v_6$. In the example, however, $FT_3^6(4) = 1$, which implies that routes from v_1, v_2 and v_3 to v_6 must use the link from v_3 to v_4.

2.4.3 Shortest Path First Routing (SPF)

The Shortest Path First Routing algorithm variants are a class of distributed routing algorithms where each router in the network participates in advertising and disseminating topology information to their neighbors, and the disseminated information is used to make routing decisions in the network. IP routing under SPF follows single path routing and destination-based forwarding, which implies that the routing continuity constraints (2.18), (2.19) and (2.20) and the forwarding constraints (2.21), (2.22) and (2.23) are applicable in this context. Automatic routing convergence within SPF also constrains the routing to be symmetric, i.e.

$$\forall v_i, v_j, v_s, v_d \in \mathcal{V} \ : \ r_{ij}^{sd} = r_{ji}^{ds} \tag{2.24}$$

Routing within SPF is constrained to follow the *shortest path* where the *length* of a path is estimated as the sum of *link weights* of links used on the specified path. As indicated in Section 2.3.1, the link weight of a link, if it exists, is fixed and is given by the constant $\hat{w}_{ij}$ $(i \neq j)$. In the model, the actual link weight is a positive variable w_{ij} $(i \neq j)$, and as defined in (2.25), is equal to $\hat{w}_{ij}$ if the link exists $(L_{ij} = 1)$ or is w_∞ (a very large positive constant) otherwise $(L_{ij} = 0)$. Using the link weight definition, the path cost from v_s to v_d can be defined as the variable RC_{sd} $(s \neq d)$, which is computed in (2.26). Here, for each link that is used, $r_{ij}^{sd} = 1$ and therefore the sum of the product $w_{ij} \cdot r_{ij}^{sd}$ over all links will give the cost of the path. This equation is non-linear, but as a route can only use a link which exists (2.16), we can substitute w_{ij} with $\hat{w}_{ij}$ to compute the route cost as shown in (2.27).

$$\forall v_i, v_j, \in \mathcal{V} \ : \ w_{ij} = L_{ij} \cdot \hat{w}_{ij} + (1 - L_{ij}) \cdot w_\infty \tag{2.25}$$

$$\forall v_s, v_d, \in \mathcal{V} \ : \ RC_{sd} = \sum_{ij} w_{ij} \cdot r_{ij}^{sd} \tag{2.26}$$

$$\forall v_s, v_d, \in \mathcal{V} \ : \ RC_{sd} = \sum_{ij} \hat{w}_{ij} \cdot r_{ij}^{sd} \tag{2.27}$$

Using the routing cost, the constraint in (2.28) ensures that only the *shortest path* can be used for a route from v_s to v_d. (2.28) constrains the route cost from v_s to v_d to be less than or equal to the routing cost to go from v_s to v_d via any v_x which is a neighbor of v_d. This presents a sufficient condition to ensure the use of the shortest path in IP routing.

$$\forall v_s, v_d \in \mathcal{V}, s \neq d : RC_{sd} \leq RC_{sx} + w_{xd} \qquad (2.28)$$

Finally, most formulations for SPF do not consider the constraints on *routing re-convergence* or the automatic re-computation of routing in the network. In the implementation of SPF routing protocols in the network, a change in a route (or routing re-convergence) is observed only when

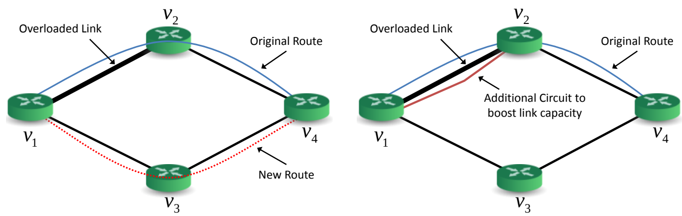
- a *shorter* path is found to the destination, or
- if links on the path currently used are not available anymore[8].

This is a critical feature of routing protocols that has been neglected in existing models. Without taking into account the conditions *when* a routing protocol can compute a new route, the solutions computed by models can compute routing configurations which may not be feasible in real networks.

An example describing the effect of the routing re-convergence constraint is presented in Fig. 2.4. Fig. 2.4(a) presents a possible solution as computed by existing ILP models, when routing re-convergence constraints are not imposed. Here, if traffic from $v_1$to v_4 uses the route $v_1 \rightarrow v_2 \rightarrow v_4$, and the link from v_1 to v_2 is overloaded, the computed solution may re-route the traffic over the alternate shortest path $v_1 \rightarrow v_3 \rightarrow v_4$. However, as specified before, most routing protocols do not automatically re-converge, and the ILP solution should instead either create a new link from v_1 to v_4 or boost the capacity of the link from v_1 to v_2 as shown in Fig. 2.4(b).

The computation of routing re-convergence requires knowledge of the original route and IP topology. The existing links in the IP network before the optimization are indicated by the boolean

[8] Traffic Engineered Shortest Path First Routing algorithms allow the re-routing of traffic onto alternate shortest paths, but are not used in practice in operator networks



(a) Solution computed without Routing re-convergence constraints

(b) Solution computed under Routing re-convergence constraint (No routing change and additional capacity installed in the network)

Figure 2.4: Example describing the need for the routing re-convergence constraints in SPF formulation

constant L_{ij}^{old} and the constant boolean $\hat{r}_{ij}^{sd}$ is used to define the routing in the IP network before the initiation of the optimization process. To identify if a change in the network topology can lead to longer or shorter routes, we use the knowledge of the routing cost before the operation $\hat{RC}_{sd}$ $(=\sum_{ij} \hat{w}_{ij} \cdot \hat{r}_{ij}^{sd})$: the condition $(\hat{RC}_{sd} - RC_{sd} \neq 0)$ implies a change in the network topology, which would lead to the re-convergence of the route in the network. The presented change only considers topology changes, where a shorter or a longer route in found in the network. However, another scenario considers a condition where the new routing cost is the same as the original routing cost, but the original path is not available in the network anymore. This scenario can be evaluated using the expression $\sum_{ij} \hat{r}_{ij}^{sd}(L_{ij}^{old} - L_{ij})$. The proposed expression is always greater than or equal to 0 and if the expression is equal to 0 the original route is available, while atleast one of the links on the original route is not available $(L_{ij} = 0)$ if the expression is greater than 0. Based on these conditions, the constraint on routing re-convergence can be expressed as

$$\forall v_s, v_d \in \mathcal{V},\ s \neq d:$$
$$P \cdot \left(| \left(\hat{RC}_{sd} - RC_{sd} \right) | + \sum_{ij} \hat{r}_{ij}^{sd} (L_{ij}^{old} - L_{ij}) \right) \geq \sum_{ij} \hat{r}_{ij}^{sd} + r_{ij}^{sd} - 2\hat{r}_{ij}^{sd} r_{ij}^{sd} \quad (2.29)$$

The expression $\sum_{ij} \hat{r}_{ij}^{sd} + r_{ij}^{sd} - 2\hat{r}_{ij}^{sd} r_{ij}^{sd}$ is equivalent to the expression $\sum_{ij} |(\hat{r}_{ij}^{sd} - r_{ij}^{sd})|$ which is greater than 0 if the original route and the computed route after optimization are not the same and is 0 otherwise. The constraint therefore enforces the condition, where if the original routing cost and the routing cost after path computation are the same ($\hat{RC}_{sd} - RC_{sd} = 0$) and if all links used in the original route are still in place ($\sum_{ij} \hat{r}_{ij}^{sd} (L_{ij}^{old} - L_{ij}) = 0$) then the final routing should be the same as the original routing. In case either of these conditions is not valid, the route from v_s to v_d can change significantly, and to ensure the same, a large positive constant P is used to ensure that the magnitude of the difference in the route costs or the number of original links not available in the network does constrain the capability to change routes, or that $P \cdot (| \left(\hat{RC}_{sd} - RC_{sd} \right) | + \sum_{ij} \hat{r}_{ij}^{sd} (L_{ij}^{old} - L_{ij}))$ is always greater than or equal to the number of links in the network.

The constraint in (2.29) is non-linear and can be linearized using the constraints in (2.30) and (2.31). The constraints compute both possible expansions for the expression $| \left(\hat{RC}_{sd} - RC_{sd} \right) |$ and one of the two are likely to indicate the positive value of the expression. The boolean variable V_{sd} and a large positive constant P_∞ ($>> P$) are used such that the positive value must satisfy the routing re-convergence constraint, while the constraint with the negative value does not have a bearing on the optimization.

$$\forall v_s, v_d \in \mathcal{V},\ s \neq d\ :$$
$$P \cdot \left(\hat{RC}_{sd} - RC_{sd}\right) + P_\infty \cdot V_{sd} + P \cdot \sum_{ij} \hat{r}_{ij}^{sd}(L_{ij}^{old} - L_{ij}) \geq \sum_{ij} \hat{r}_{ij}^{sd} + r_{ij}^{sd} - 2\hat{r}_{ij}^{sd} r_{ij}^{sd} \quad (2.30)$$

$$\forall v_s, v_d \in \mathcal{V},\ s \neq d\ :$$
$$P \cdot \left(RC_{sd} - \hat{RC}_{sd}\right) + P_\infty \cdot (1 - V_{sd}) + P \cdot \sum_{ij} \hat{r}_{ij}^{sd}(L_{ij}^{old} - L_{ij}) \geq \sum_{ij} \hat{r}_{ij}^{sd} + r_{ij}^{sd} - 2\hat{r}_{ij}^{sd} r_{ij}^{sd} \quad (2.31)$$

2.5 Routing of Dynamic Optical Circuits

The model presented here focuses primarily on the effect of dynamic optical circuits on IP routing, which is determined by the placement of optical circuits. This section presents constraints to model the routing of dynamic optical circuits in a generic optical transport network.

The topology of the optical transport network is described by the graph $\mathcal{G}^{\mathcal{O}}(\mathcal{V}^{\mathcal{O}}, \mathcal{E}^{\mathcal{O}})$ consisting of switches $v_i^O \in \mathcal{V}^{\mathcal{O}}$ and directed links $e_{ij}^O \in \mathcal{E}^{\mathcal{O}}$ with available capacity C_{ij}^{OT} . As specified before, the router v_x is connected with switch $v_x^O \in \mathcal{V}^{\mathcal{O}}$, and therefore circuits of type $t \in \mathcal{T}$ from v_x to v_y are routed from v_x^O to v_y^O, with the number of these circuits given by X_{xy}^t.

A positive integer variable $RO_{ij}^{xy}(t)$ $(x \neq y,\ i \neq j)$ is used to indicate the routing of optical circuits, and defines the number of circuits of type $t \in \mathcal{T}$ that are reserved on the link e_{ij}^O for connections from v_x^O to v_y^O. Routing in the optical network is subject to the routing continuity constraints given by

$$\forall t \in \mathcal{T}, v_x^O, v_y^O \in \mathcal{V}^{\mathcal{O}} \; : \; \sum_i RO_{xi}^{xy}(t) = \sum_i RO_{iy}^{xy}(t) \; = \; X_{xy}^t \quad (2.32)$$

$$\forall t \in \mathcal{T}, v_x^O, v_y^O \in \mathcal{V}^{\mathcal{O}} \; : \; \sum_i RO_{ix}^{xy}(t) = \sum_i RO_{yi}^{xy}(t) \; = \; 0 \quad (2.33)$$

$$\forall t \in \mathcal{T}, v_x^O, v_y^O, v_i^O \in \mathcal{V}^{\mathcal{O}}, i \neq x, y :$$
$$\sum_{k: e_{ki}^O \in \mathcal{E}^{\mathcal{O}}} RO_{ki}^{xy}(t) = \sum_{j: e_{ij}^O \in \mathcal{E}^{\mathcal{O}}} RO_{ij}^{xy}(t) \leq X_{xy}^t \quad (2.34)$$

$$\forall t \in \mathcal{T}, v_x^O, v_y^O \in \mathcal{V}^{\mathcal{O}}, e_{ij}^O \in \mathcal{E}^{\mathcal{O}} \; :$$
$$\sum_{xy} \sum_t RO_{ij}^{xy}(t) \cdot C_t^O \leq C_{ij}^{OT} \quad (2.35)$$

The constraints (2.32), (2.33) and (2.34) are routing continuity constraints. (2.32) constraints the total number of dynamic optical circuits of $t \in \mathcal{T}$ from v_x^O to v_y^O using an outgoing link at v_x^O and an incoming link at v_y^O are equal to X_{xy}^t. (2.33) ensures that no optical circuits terminate at their source or emerge from their destination, while (2.34) ensures routing continuity at intermediate optical switches.

The routing of optical circuits is additionally constrained by the available capacity on the links in the optical network using (2.35).

The model presented here is a generic model for the routing of multiple dynamic optical circuits from a source to a destination, and technology-specific constraints such as Optical Signal to Noise Ratio (OSNR) and wavelength continuity constraints for WDM, as well as specific constraints on the use of different routing paths for multiple circuits between a source and a destination can be incorporated in this model [33, 34].

2.6 Formulation Under Unknown Traffic Matrix Conditions

The formulation presented in the previous sections assumes that complete traffic matrix information is known. As a result, the link capacity constraint (2.17) uses an expression of the form

$$\sum_{sd} a_{ij}^{sd} \cdot \hat{\lambda}_{sd} \leq \alpha \cdot C_{ij} \tag{2.36}$$

where a_{ij}^{sd} is a boolean[9] variable that indicates if the traffic from v_s to v_d is routed over the link from v_i to v_j. However, in case the traffic matrix is not known (2.37), the parameter $\hat{\lambda}_{sd}$ is not known and must be replaced by the variable λ_{sd} making the expression non-linear.

$$\sum_{sd} a_{ij}^{sd} \cdot \lambda_{sd} \leq \alpha \cdot C_{ij} \tag{2.37}$$

The solutions presented here makes use of readily available traffic measurements and guarantees bounds on aggregate traffic, which ensures that the solutions never under-estimate the traffic on any link in the network. Traffic measurements from IP networks such as IP link load measurements $LinkLoad_{ij}$ and virtual output queue measurements γ_{ik}^{j} can be represented as a linear expressions of sum over the variable λ_{sd} with boolean coefficients, indicating if the traffic between a particular source-destination router pair contributes to the measurement.

[9] The choice of boolean coefficients for traffic is driven from the assumption that the IP network has no routing loops and uses a single unique path from a source to a destination.

$$\forall v_i, v_j \in \mathcal{V}, \hat{L}_{ij} = 1 \;:\; \sum_{sd} \lambda_{sd} \cdot \hat{r}_{ij}^{sd} = LinkLoad_{ij} \tag{2.38}$$

$$\forall v_i, v_j, v_k \in \mathcal{V}, \hat{L}_{ij} = \hat{L}_{jk} = 1 \;:\; \sum_{sd} \lambda_{sd} \cdot \hat{r}_{ij}^{sd} \cdot \hat{r}_{jk}^{sd} = \gamma_{ik}^{j} \tag{2.39}$$

$$\forall e_{ij} \in E \;:\; \sum_{d} \lambda_{id} \cdot \hat{r}_{ij}^{id} = \gamma_{ij}^{i} \tag{2.40}$$

$$\forall e_{ij} \in E \;:\; \sum_{s} \lambda_{sj} \cdot \hat{r}_{ij}^{sj} = \gamma_{ij}^{j} \tag{2.41}$$

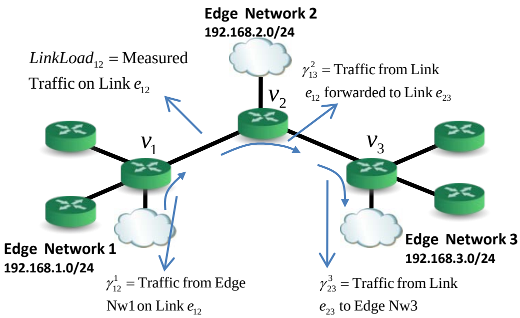


Figure 2.5: Traffic measurements available in IP networks [1]

The different cases of the link load and virtual output queue measurements are shown in Fig. 2.5. Here, traffic measured on the link from v_1 to v_2 is given by $LinkLoad_{12}$, while traffic on the link from v_1 to v_2 which is forwarded to v_3 is given by γ_{13}^{2}. The figure also presents two distinct cases of virtual output queue measurements made at the source and destination of traffic. For example, γ_{12}^{1} indicates traffic entering the network from v_1 with the next hop as v_2 and γ_{23}^{3} indicates the traffic exiting the network at router v_3 with the previous hop as router v_2.

In this model, the available measurements are used to generate a measurement expression set D , consisting of unique measurement expressions D_i represented as

$$D_i \ : \ \sum_{sd} d^i_{sd} \cdot \lambda_{sd} = B_i \tag{2.42}$$

where the boolean coefficient d^i_{sd} indicates if λ_{sd} contributes to the measured traffic B_i . The uniqueness constraint enforces that no two expressions in D have exactly the same values for all the boolean coefficients d^i_{sd} or

$$\sum_{sd} |d^i_{sd} - d^j_{sd}| = 0 \ iff. \ i = j$$

The uniqueness constraint implies that some duplicate measurements gathered from the network may only be treated as a single expression in the set. For example, in the network in Fig. 2.5, the expression generated using the measurement for $LinkLoad_{12}$ $(= \lambda_{12} + \lambda_{13})$ would have $d^1_{12} = d^1_{13} = 1$ and all other $d^1_{xy} = 0$, which is exactly the same as the expression for γ^1_{12} and are not treated as different expressions in D .

In this model, in an effort to *linearize* (2.37), additional constraints are used to check if the Left Hand Side (LHS) of (2.37) for a particular IP link matches a known expression $D_i \in D$. In case an exact match D_i is found, the LHS of (2.37) can be replaced by B_i . In case no matches are found, upper bounds computed on λ_{sd} (λ^{max}_{sd}) are used to compute the worst case upper bound, and the expression $\sum_{sd} a^{sd}_{ij} \lambda^{max}_{sd}$ is used to replace the original expression.

The linearization uses two boolean variables S^x_{ij} and S_{ij} $(i \neq j)$, with S^x_{ij} indicating if the link load expression on link from v_i to v_j is the same as the expression $D_x \in D$ using (2.43) and (2.44). (2.43) constrains $S^x_{ij} = 1$ if any $d^x_{sd} \in D_x$ is not the same as the corresponding a^{sd}_{ij}, while (2.44) constrains $S^x_{ij} = 0$ in case all $d^x_{sd} = a^{sd}_{ij}$. Both constraints are non-linear, and using the fact that both a^{sd}_{ij} and d^x_{sd} are boolean (making $|a^{sd}_{ij} - d^x_{sd}|$ also boolean), the constraints are linearized as (2.45) and (2.46) respectively.

$$\forall D_x \in D, v_i, v_j \in \mathcal{V}$$

$$\forall v_s, v_d \in \mathcal{V} \;:\; S_{ij}^{x} \geq |a_{ij}^{sd} - d_{sd}^{x}| \tag{2.43}$$

$$S_{ij}^{x} \leq \sum_{sd} |a_{ij}^{sd} - d_{sd}^{x}| \tag{2.44}$$

$$\forall v_s, v_d \in \mathcal{V} \;:\; S_{ij}^{x} \geq a_{ij}^{sd} + d_{sd}^{x} - 2 \cdot d_{sd}^{x} \cdot a_{ij}^{sd} \tag{2.45}$$

$$S_{ij}^{x} \leq \sum_{sd} \left(a_{ij}^{sd} + d_{sd}^{x} - 2 \cdot d_{sd}^{x} \cdot a_{ij}^{sd} \right) \tag{2.46}$$

Based on these constraints, S_{ij}^{x} identifies if a particular expression $D_x \in D$ matches the link load expression. On the other hand, S_{ij} identifies if no expression in D matches the link load expression using the constraint (2.47). This constraint uses the fact that all expressions in the set D are unique and therefore only one expression in D can match the link load expression at any time. Based on this constraint, $S_{ij} = 0$ when none of the expressions in D match the link load expression ($S_{ij}^{x} = 1 \;\forall D_x \in D$) and $S_{ij} = 1$ otherwise.

$$\forall v_i, v_j \in \mathcal{V} \;:\; S_{ij} = \sum_{x} \left(1 - S_{ij}^{x} \right) \tag{2.47}$$

Using the indicator variables S_{ij} and S_{ij}^{x} , the generic non-linear constraint for IP link load (2.37) can be replaced by the constraints

$$\forall v_i, v_j \in \mathcal{V}, i \neq j \;:\; \sum_{x} (1 - S_{ij}^{x}) \cdot B_x \leq \alpha \cdot C_{ij} \tag{2.48}$$

$$\forall v_i, v_j \in \mathcal{V}, i \neq j \;:\; \sum_{sd} a_{ij}^{sd} \cdot \lambda_{sd}^{max} - S_{ij} \cdot \lambda_{\infty} \leq \alpha \cdot C_{ij} \tag{2.49}$$

(2.48) checks if any of the traffic expressions match the link load expression, and in case an expression matches (S_{ij}^{x} =0), the LHS of (2.48) is equal to the measured value B_i . In case an expression matches, $S_{ij} = 1$ and a large positive traffic value λ_{∞} is used to render the constraint (2.49) redundant. However, if none of the expressions match ($S_{ij}^{x} = 1 \;\forall D_x \in D$, $S_{ij} = 0$), (2.48) is redundant, while the constraint in (2.49) uses the traffic bounds λ_{sd}^{max} in the link load expression.

This constraint can be adapted to the SPF mechanism by mapping the coefficients a_{ij}^{sd} to the corresponding coefficients in the link load expressions (2.17), or

$$SPF \ : \ a_{ij}^{sd} = r_{ij}^{sd} \tag{2.50}$$

The performance of this approach is dependent on the mechanism used to generate the expression set D, as the optimizer attempts to compute solutions where the routing configurations in the network are such that traffic expressions in D can be found for most (if not all) IP links. Therefore, if only link load measurements are used, the solution computed would guarantee a solution where optical circuits are primarily used to increase the capacity of existing links and routing in the network is not affected. In the numerical study presented in the next chapter, the model also include measurements for γ_{ik}^{j} and expressions obtained by subtracting the traffic measured on the virtual output queues from the link loads, i.e. $LinkLoad_{ij} - \gamma_{ik}^{j}$ and $LinkLoad_{jk} - \gamma_{ik}^{j}$. These expressions can possibly be used to compute traffic for two-hop optical circuits under the different mechanisms. For example, for the topology in Fig. 2.5, in case an optical circuit is established from v_1 to v_3, the traffic on this new link is given by γ_{13}^{2} while the remaining traffic on the links from v_1 to v_2 and from v_2 to v_3 are given by $LinkLoad_{12} - \gamma_{13}^{2}$ and $LinkLoad_{23} - \gamma_{13}^{2}$ respectively.

2.7 Summary

This chapter highlighted the limitations of the existing IP-optical proposals in modeling the effect of optical circuits on IP routing, and presented an ILP based framework that can be applied to a variety of IP routing scenarios. The model identified critical features including *destination-based forwarding* and *routing re-convergence* that have not been studied in existing research, and proposed novel constraints for the same. Each of these constraints can significantly affect the performance of the model in terms of the required optical capacity, which is further highlighted in the numerical study presented in the next chapter.

The chapter presented a *new optimization objective* that can simultaneously compute the location and placement of new dynamic optical circuits and the decommissioning of existing dynamic optical circuits. This is a critical requirement for application in a dynamic network scenario, but has not been studied to date.

The chapter also presented a unique formulation for computing the requirements on dynamic optical circuits under unknown traffic matrix conditions. The proposed formulation is designed to use easily available traffic measurements from the network which is critical when applying the model in real network scenarios. The model is also unique as it provides deterministic performance guarantees in terms of required optical capacity, which are equal to the scenario where the capacity of IP links are adjusted to the point where the link utilization lower than the threshold α.

3

New Approach for IP-Optical Integration - Optical Bypass

The use of traditional IP routing schemes when introducing dynamic optical circuits requires complex computation to evaluate the location and placement of optical circuits, which is complicated further when the complete IP traffic matrix is not known. As a result most deployments of dynamic optical circuits in IP networks today employ end-to-end circuits which are typically provisioned for specific applications with very high bandwidth demands, while the routing and capacity allocation for IP traffic remains static.

This chapter presents the *Optical Bypass (BY)* scheme, which is a novel approach proposed in this thesis for introducing optical circuits in IP networks. The Optical Bypass scheme was designed to address some of the management and computation challenges associated with the use of traditional IP routing schemes. The proposed scheme uses hidden bypasses in the middle of the network to re-route IP traffic away from overloaded IP links/route segments. The Optical Bypass scheme is designed for application in dynamic network scenarios, and therefore the formulation presented has significantly lower time complexity and affects a very small number of IP routes.

The model is also well suited computing solutions under unknown traffic matrix conditions using the formulation presented in the previous chapter. The results presented in this chapter show that the proposed approach requires lesser optical capacity than the traditional SPF approach while reducing the number of routing changes under known and unknown traffic matrix conditions.

This chapter presents the motivation behind the Optical Bypass mechanism, followed by the formulation constrained by the network model presented in the previous section. The chapter then goes on to present a numerical study which compares the performance of the BY schemes with the traditional ER and SPF schemes.

3.1 Supporting Publications

1. M. Chamania, A. Jukan, "A Comparative Analysis of the Effects of Dynamic Optical Circuit Provisioning on IP Routing," to appear in **IEEE/ACM Transactions on Networking**, 2013.

2. M. Chamania, M. Caria, A. Jukan, "Achieving IP Routing Stability with Optical Bypass," **Optical Switching and Networking**, vol. 7, no. 4, pp. 173-184, 2010.

3. M. Chamania, M. Caria, A. Jukan, "A Comparative Performance Analysis of IP Traffic Offloading Schemes over Dynamic Circuits," **IEEE INFOCOM**, 2011.

4. M. Chamania, M. Caria, A. Jukan, "Effective Usage of Dynamic Circuits for IP Routing," **IEEE International Conference on Communications (ICC)**, 2010.

5. M. Chamania, M. Caria, A. Jukan, "Achieving IP Routing Stability with Optical Bypass," **IEEE Advanced Networks and Telecommunication Systems (ANTS)**, 2009.

3.2 Motivation

As outlined in Section 2.2, the introduction of dynamic optical circuits under legacy IP routing schemes presents significant challenges in terms of evaluating the placement and capacity of optical circuits, and can also lead to large-scale routing changes in the network which are not desirable. As a result, only science networks have deployed dynamic optical circuits in conjunction with IP networks using frameworks such as Lambdastation [2] and Phoebus [11]. A scenario describing the use of the Lambdastation framework in a multi-layer IP-over-optical network is presented in Fig. 3.1. Here, regular traffic between the sites A and B is routed over the IP network (1). However, a specific application requiring high-bandwidth connectivity between the sites can request the central Lambdastation controller for a circuit between the sites (2). If possible, the Lambdastation controller provisions an end-to-end circuit between the sites, and configures routers near the edge of sites A and B to re-route application traffic[1] onto the established circuit using policy-based routing rules [44] (3). As a result, application traffic is now routed over the established circuit (4), while the rest of the traffic between the sites traverses the IP network. The primary reasons behind the popularity of such frameworks are the deterministic nature of the demands for high-capacity circuits (in terms of required capacity and running-time) and the fact that the establishment of the circuit does not affect the IP topology in the core network. Given its success, a class of approaches proposed monitoring and automatically identifying high-capacity flows in commercial networks, and *offloading* these connections onto end-to-end optical circuits in order to reduce the traffic on intermediate routers in the network [45, 46]. Internet traffic in commercial domains, however, contains numerous

[1] Application traffic in this scenario can be identified using the unique tuple *<Source IP Address, Source Port, Destination IP Address, Destination Port>*

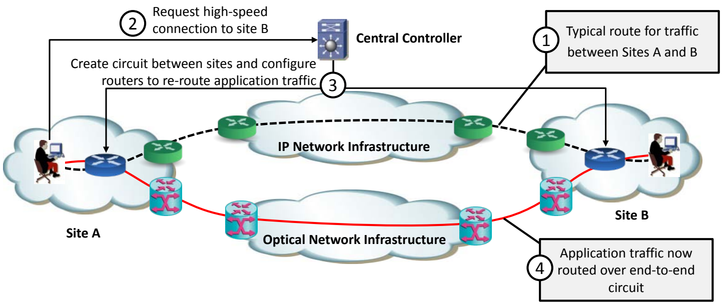


Figure 3.1: Typical Deployment of IP-Optical networks with dynamic optical circuits (Based on the LambdaStation [2] Deployment scenario)

low-capacity and relatively short-lived application flows and as a result, solutions proposing end-to-end circuits for application flows are not directly applicable in commercial networks.

3.3 The Optical Bypass (BY) Approach

Taking into account the shortcomings of current proposals for introducing optical circuits in IP networks, the Optical Bypass mechanism is proposed in this thesis, which advocates the need to introduce optical circuits in IP networks with minimum impact on IP routing [47, 48, 37]. The Optical Bypass (BY) is a novel solution that is designed to

- Allow the dynamic introduction of optical capacity
- Ensure that change in IP routing and consequently the associated administrative overhead is small

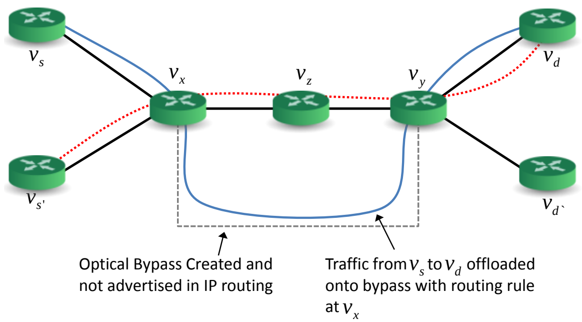


Figure 3.2: Example describing the use of Optical Bypass in a network

- Have low computational complexity to be applicable in real-time

In order to address the aforementioned requirements, the Optical Bypass mechanism uses optical circuits as hidden bypasses that are not advertised in the IP routing protocol. By not advertising new IP links in the IP routing protocol, the Optical Bypass mechanism ensures that routing protocols such as OSPF remain stable and do not re-converge to a new routing configuration. The optical bypasses are established across overloaded links, and specific forwarding rules (e.g. static routing or policy-based routing [44]) are established at the ingress of the bypass to divert traffic from the overloaded IP links onto the optical bypasses. An example of the same can be seen in Fig. 3.2. Here a dynamic optical circuit is established from v_x to v_y but is not advertised in the IP routing protocol, and traffic from v_s to v_d is offloaded onto this bypass by establishing a routing rule at router v_x. As a result, traffic from $v_{s'}$ to v_d uses the original routing path, and only traffic from v_s to v_d is offloaded onto the bypass.

In the Optical Bypass mechanism, as specified in Section 2.3.2, optical circuits can either be used to boost the capacity of existing IP links or can be used to create new IP links that are used as *bypasses*. When an optical circuit is used to boost the capacity of an existing link, the link weight metric used by SPF routing protocols remains unchanged as defined in Section 2.3.2, thereby ensuring that routing protocols do not re-converge. When establishing *bypasses*, many different mechanisms can be used to ensure that routing protocols are not affected by the same, some of which include

- Assigning a very high link weight metric to the *bypass* so that the SPF routing protocol does not use the *bypass* for routing traffic
- SPF routing areas are defined using IP subnets, and assigning the endpoints of the bypass IP addresses outside this subnet ensures that the routing protocol does not *see* the new IP link.

3.4 Formulation of the Optical Bypass

This section presents the formulation of the BY solution under the network model constraints and optimization objectives presented in Section 2.3.2. Specifically, the optimization is subject to the same optimization objective (2.12), as well as constraints on link capacity (2.8) and constraints on optical circuits (2.1), (2.2), (2.3).

Within the formulation, optical circuits between previously non-neighboring IP routers are treated as optical bypasses, and instead of the formulation presented in the previous chapter to identify the location of an IP link, (3.1) ensures that all optical bypasses established in the IP network are bi-directional (but can have asymmetric upstream/downstream capacities).

$$\forall v_x, v_y \in \mathcal{V}, \hat{L}_{xy} = 0 \ : \ X_{xy} = X_{yx} \tag{3.1}$$

The mechanism for forwarding traffic onto an optical bypass is dependent on the forwarding capabilities available on the routers in the network, and presents two formulations in regards to the same. The Optical Bypass (BY) formulation assumes that routers at the ingress of the optical bypass can create forwarding rules based on both the source and destination addresses and is presented in Section 3.4.1. In the case of typical IP routers which are optimized for destination-based forwarding only, the Optical Bypass with Destination-Based Forwarding (BY-D) formulation presented in Section 3.4.2 allows the forwarding of traffic onto optical bypasses based only on the destination of the traffic.

3.4.1 Problem Formulation - Optical Bypass (BY)

As described in the previous section, the Optical Bypass model can offload traffic on a dynamic optical circuit across two or more hops under the constraints that the ingress and egress of the optical circuit should lie on the original routing path. For formulating the ILP under these constraints, two boolean routing constants: ψ_{xy}^{sd} and $\psi_{xy}^{sd}(ij)$ are used. The constant ψ_{xy}^{sd} indicates if the route for traffic from v_s to v_d has v_x and v_y as intermediate hops. Unlike the original routing constant $\hat{r}_{ij}^{sd}$, where router v_i and v_j must be connected by an IP link, an IP interconnection between v_x and v_y is not required for $\psi_{xy}^{sd} = 1$.

Using the routing constant ψ_{xy}^{sd}, it is possible to identify if an optical bypass from v_x to v_y can be used to offload traffic from v_s to v_d. Once offloaded, however, the traffic on links in the route segment from v_x to v_y will be affected, and must be accounted for in the formulation. The second routing constant $\psi_{xy}^{sd}(ij)$ indicates if the IP link from v_i to v_j belongs to the route segment from v_x to v_y which in turn is a part of the route from v_s to v_d, and is used to identify the affected links when traffic from v_s to v_d is offloaded on an optical bypass from v_x to v_y.

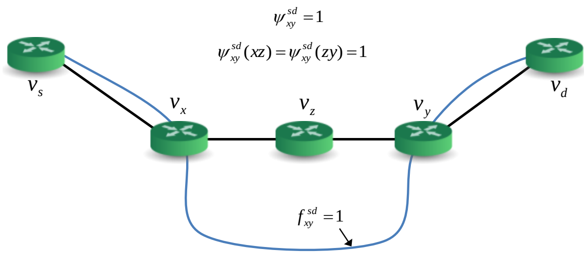


Figure 3.3: Routing Constants and Variables used for formulating the Optical Bypass Model

An example of the same can be seen in Fig. 3.3, where the original route for traffic from v_s to v_d is $v_s \to v_x \to v_z \to v_y \to v_d$. The routers v_x and v_y lie on the original routing path from v_s to v_d and hence $\psi_{xy}^{sd} = 1$. Furthermore, in case a bypass is established from v_x to v_y, the links from v_s to v_x and from v_y to v_d are not affected, but the links traversed in the segment from v_x to v_y (v_x to v_z and v_z to v_y) are affected, and the parameter values $\psi_{xy}^{sd}(xz) = 1$ and $\psi_{xy}^{sd}(zy) = 1$ are used to identify the same.

Using the routing information constants ψ_{xy}^{sd} and $\psi_{xy}^{sd}(ij)$, the BY problem is formulated as presented below. A binary offloading variable f_{xy}^{sd} ($s \neq d$, $x \neq y$) is used to indicate if traffic from v_s to v_d is offloaded over an optical bypass from v_x to v_y. In the formulation, dynamic optical circuits established in parallel to existing IP links are used to boost the IP link capacity, and therefore only dynamic optical circuits established between previously non-neighboring IP routers are used as optical bypasses.

$$\forall v_s, v_d, v_x, v_y \in \mathcal{V},\ \hat{L}_{xy} = 0\ :\ f_{xy}^{sd} \leq X_{xy} \tag{3.2}$$

$$\forall v_s, v_d, v_x, v_y \in \mathcal{V},\ \hat{L}_{xy} = 0\ :\ f_{xy}^{sd} \leq \psi_{xy}^{sd} \tag{3.3}$$

$$\forall v_s, v_d, v_i, v_j \in \mathcal{V},\ \hat{L}_{ij} = 1\ :\ \sum_{xy:\hat{L}_{xy}=0} \psi_{xy}^{sd}(ij) \cdot f_{xy}^{sd} \leq 1 \tag{3.4}$$

The constraint in (3.2) ensures that traffic between any source-destination router pair can only be offloaded from v_x to v_y in case a) no existing link exists between v_x and v_y ($\hat{L}_{xy} = 0$) and b) at least one dynamic optical circuit is established between v_x and v_y ($X_{xy} = 1$). (3.3) further ensures that traffic from v_s to v_d can only be offloaded onto a bypass from v_x to v_y if and only if the routers v_x and v_y belong to the original routing path from v_s to v_d ($\psi_{xy}^{sd} = 1$).

The formulation does not constrain the number of times traffic from v_s to v_d is offloaded onto an optical bypass. However, it is necessary to ensure that no overlapping optical bypasses are chosen to offload the same traffic. The constraint in (3.4) is used to ensure that no overlapping bypasses are used to offload the same traffic. The constraint is applied to all existing IP links in the network ($\hat{L}_{ij} = 1$) and all possible unique source-destination router pairs. The expression $\psi_{xy}^{sd}(ij) \cdot f_{xy}^{sd} = 1$ if traffic from v_s to v_d uses the optical bypass from v_x to v_y ($f_{xy}^{sd} = 1$) and if the intermediate route segment from v_x to v_y contains the link from v_i to v_j ($\psi_{xy}^{sd}(ij) = 1$). Ensuring that the sum of this expression over all possible bypasses is less than or equal to 1 guarantees that a link can only be bypassed once for traffic from v_s to v_d, thereby ensuring that no overlapping optical bypasses are used by the formulation. An example of the same can be described based on Fig. 3.3. Here, traffic from v_s to v_d cannot simultaneously use bypasses from v_x to v_y and from v_z to v_d. When (3.4) is applied on the link from v_z to v_y, $\psi_{xy}^{sd}(zy) = \psi_{zd}^{sd}(zy) = 1$ and therefore only one of f_{xy}^{sd} and f_{zd}^{sd} can be equal to 1.

The constraints (3.2), (3.3) and (3.4) and necessary and sufficient conditions for ensuring that traffic from v_s to v_d uses a valid single-path route from the source to the destination. Apart from these, link capacity constraints are enforced on existing IP links as well as optical bypasses established in the system.

For established optical bypasses, the link capacity constraint is defined as:

$$\forall v_x, v_y \in \mathcal{V},\ \hat{L}_{xy} = 0 : \sum_{sd} f_{xy}^{sd} \cdot \hat{\lambda}_{sd} \leq \alpha \cdot C_{xy} \tag{3.5}$$

The variable f_{xy}^{sd} indicates if traffic from v_s to v_d with magnitude $\hat{\lambda}_{sd}$ is offloaded onto the optical bypass, and the constraint ensures that the sum of all traffic offloaded onto the optical bypass does not violate the maximum link utilization threshold α.

For already existing IP links, the link capacity constraint is presented in (3.6). The constraint is designed so that the original routing information $\hat{r}_{ij}^{sd}$ is used to identify if traffic from v_s to v_d traverses the link from v_i to v_j while the offloading variable f_{xy}^{sd} is used to check if traffic from v_s to v_d is offloaded across a segment v_x to v_y that traverses the link from v_i to v_j.

$$\forall v_i, v_j \in \mathcal{V},\ \hat{L}_{ij} = 1 \ :$$
$$\sum_{sd} \hat{\lambda}_{sd} \cdot \hat{r}_{ij}^{sd} \left(1 - \sum_{xy:\hat{L}_{xy}=0} \psi_{xy}^{sd}(ij) \cdot f_{xy}^{sd} \right) \leq \alpha \cdot C_{ij} \tag{3.6}$$

Here, as indicated before, the expression $\hat{\lambda}_{sd} \cdot \hat{r}_{ij}^{sd}$ is used to indicate the known contribution of traffic from v_s to v_d on the link from v_i to v_j while the expression $\left(1 - \sum_{xy:\hat{L}_{xy}=0} \psi_{xy}^{sd}(ij) \cdot f_{xy}^{sd} \right)$ is a boolean condition derived from (3.4). The constraint in (3.4) ensures that only one bypass can be used to offload traffic from v_s to v_d over an existing link from v_i to v_j, at which point $\sum_{xy:\hat{L}_{xy}=0} \psi_{xy}^{sd}(ij) \cdot f_{xy}^{sd} = 1$, which makes the boolean expression in (3.6) equal to 0, thus eliminating the contribution of the traffic from v_s to v_d from the link in question.

3.4.2 Problem Formulation - Optical Bypass with Destination-Based Forwarding

The formulation presented in the previous section assumes that traffic can be offloaded onto optical bypasses based on both the source and destination of traffic. However, traditional IP forwarding only supports destination-based forwarding rules and policy-based implementations such as [44] on traditional routers are not designed for high traffic loads observed in core networks. The Optical Bypass with Destination-Based Forwarding (BY-D) formulation presents an approach where traffic is bypassed only based on the destination address. In the formulation, there is an inherent assumption that all routing in the network is based on destination-based forwarding. Here, when an optical bypass is established from v_x to v_y, the offloading variable f^d_{xy} $(x \neq y,\ d)$ indicates if traffic to destination v_d from v_x and all routers upstream from v_x is offloaded onto the optical bypass.

The constraints on the forwarding of traffic onto the optical bypass in this scenario is similar to the formulation presented in the previous section. (3.7) ensures that for traffic to be offloaded onto a bypass from v_x to v_y, at least one dynamic optical circuit must be established between the routers in question. The constraint in (3.8) ensures that traffic to v_d can only be offloaded over a bypass from v_x to v_y in case the segment from v_x to v_y lies on the original route. Here, sources of traffic consist of the router v_x and all routers upstream from v_x, and under the destination-based forwarding, traffic from all these sources to v_d use the same downstream route from v_x. As a result, only the consideration of v_x as the source of traffic (ψ^{xd}_{xy}) is sufficient for enforcing the constraint in (3.8). Finally, (3.9) uses a formulation similar to that used in (3.4) to ensure that traffic from v_s to v_d is not offloaded over overlapping optical bypasses.

$$\forall v_d, v_x, v_y \in \mathcal{V},\ \hat{L}_{xy} = 0\ :\ f^d_{xy} \leq X_{xy} \tag{3.7}$$

$$\forall v_d, v_x, v_y \in \mathcal{V},\ \hat{L}_{xy} = 0\ :\ f^d_{xy} \leq \psi^{xd}_{xy} \tag{3.8}$$

$$\forall v_s, v_d, v_i, v_j \in \mathcal{V},\ \hat{L}_{ij} = 1\ :\ \sum_{xy:\hat{L}_{xy}=0} \psi^{sd}_{xy}(ij) \cdot f^d_{xy} \leq 1 \tag{3.9}$$

The link capacity constraints for the BY-D formulation are also similar to those seen in the BY formulation, and are presented below. The link capacity constraint for the established bypasses (3.10) uses the expression $\psi^{sd}_{xy} \cdot f^d_{xy}$ to identify if the traffic from v_s to v_d uses the segment from v_x to v_y, while the variable f^d_{xy} indicates if the traffic to v_d is offloaded onto the bypass from v_x to v_y at v_x.

$$\forall v_x, v_y \in \mathcal{V},\ \hat{L}_{xy} = 0 : \sum_{sd} \psi^{sd}_{xy} \cdot f^d_{xy} \cdot \hat{\lambda}_{sd} \leq \alpha \cdot C_{xy} \tag{3.10}$$

The capacity constraint for existing IP links is given by (3.11) and is similar to the constraint presented in (3.6). This constraint uses a boolean expression $\left(1 - \sum_{xy:\hat{L}_{xy}=0} \psi^{sd}_{xy}(ij) \cdot f^d_{xy}\right)$ derived from (3.9) to identify if traffic from v_s to v_d has been offloaded away from the link from v_i to v_j, while the $\hat{r}^{sd}_{ij}$ parameter is used to determine if traffic from v_s to v_d was originally routed on the link from v_i to v_j.

$$\forall e_{ij} \in E,\ \hat{L}_{ij} = 1\ :$$
$$\sum_{sd} \hat{\lambda}_{sd} \cdot \hat{r}^{sd}_{ij} \left(1 - \sum_{xy:\hat{L}_{xy}=0} \psi^{sd}_{xy}(ij) \cdot f^d_{xy}\right) \leq \alpha \cdot C_{ij} \tag{3.11}$$

3.4.3 Formulation Under Unknown Traffic Matrix Conditions

In order to compute solutions under unknown traffic matrix conditions, the link load constraints for the BY and BY-D mechanisms are adapted to the formulation presented in Section 2.6. Unlike the ER and SPF formulation, however, the link load constraints for the Optical Bypass formulations differ when applied to existing links ($\hat{L}_{ij} = 1$) or to optical bypasses ($\hat{L}_{ij} = 0$). As a result, the variable a_{ij}^{sd} in the case of BY and BY-D formulations is given by:

$$BY$$

$$\hat{L}_{ij} = 1 \ : \ a_{ij}^{sd} = \hat{r}_{ij}^{sd} \left(1 - \sum_{xy:\hat{L}_{xy}=0} \psi_{xy}^{sd}(ij) \cdot f_{xy}^{sd} \right) \tag{3.12}$$

$$\hat{L}_{ij} = 0 \ : \ a_{ij}^{sd} = f_{xy}^{sd} \tag{3.13}$$

$$BY - D$$

$$\hat{L}_{ij} = 1 \ : \ a_{ij}^{sd} = \hat{r}_{ij}^{sd} \left(1 - \sum_{xy:\hat{L}_{xy}=0} \psi_{xy}^{sd}(ij) \cdot f_{xy}^{d} \right) \tag{3.14}$$

$$\hat{L}_{ij} = 0 \ : \ a_{ij}^{sd} = f_{xy}^{d} \cdot \psi_{xy}^{sd} \tag{3.15}$$

The proposed formulation in Section 2.6 is well-suited for application with the Optical Bypass mechanism, where the traffic on a link is not affected unless it is bypassed, and hence, it is likely that a matching expression for the traffic on this link is found in the expression set D. Also, in case of two-hop bypasses, the expressions γ_{ik}^{j}, $LinkLoad_{ij} - \gamma_{ik}^{j}$ and $LinkLoad_{jk} - \gamma_{ik}^{j}$ accurately estimate the traffic on the optical bypass as well as the residual traffic on the IP links that were bypassed.

3.5 Numerical Evaluation

The evaluation presented in this section analyzes the performance of the explicit routing based ER and ER-D schemes, the SPF scheme, and the optical bypass schemes BY and BY-D. The evaluation also presents computations under unknown traffic matrix conditions for the SPF, BY and BY-D schemes, termed as SPF-NoTM, BY-NoTM and BY-D-NoTM respectively. The results of this analysis have been previously presented in [1].

The performance of different schemes is evaluated based on the optical capacity installed ($\sum X^t_{xy} \cdot C^O_t$) and effect on routing (and consequently traffic/network services). For the optimization objective, the parameter $Cost^t_{xy}$ is given by

$$Cost^t_{xy} = IFCost^t + HopCount_{xy} * CostPerHop^t \tag{3.16}$$

where $IFCost^t$ is the cost of the interfaces used, and $CostPerHop^t$ is the cost of creating an optical circuit of type $t \in \mathcal{T}$ across a single hop. The study presented uses three different types of circuit granularities, which are presented in Table 3.1. The interface costs have been modeled based on IP interface prices (extracted from [49]) and follow a common trend where cost of high-capacity interface such as a 40G interface is typically 3-3.5 time the cost of a 10G interface and so on. The primary goal of this study is to evaluate the optical circuit placement and corresponding routing changes in the different schemes, and consequently the routing of circuits in the optical transport network is not considered, and the $CostPerHop^t$ is chosen to be an order of magnitude smaller than $IFCost^t$, making $IFCost^t$ the primary contributing factor, but still placing some extra cost for making longer circuits. In case $CostPerHop^t = 0$, while the total optical capacity installed would not be affected, the location and placement of optical circuits could be significantly different, and computed solutions may not be practical in real-life scenarios. It should also be noted here that the formulation in general is not very sensitive to small changes in cost of interfaces. However, if the cost of interfaces were to change dramatically, the computed solution can be affected significantly. For example, if high-capacity interface costs were lowered by approximately 50%, the formulation may present solutions with significantly higher installed capacity but with lower actual cost.

Bandwidth (Gbps)	$IFCost^t$	$CostPerHop^t$
2.5	10	1
10	30	3
40	90	9

Table 3.1: Interface types and associated parameters used for the numerical evaluation

The ILP was computed using the Gurobi Optimizer v5.0 [50]. All computations under 30 minutes were solved optimally with a MIP gap of $1e^{-4}$, while longer simulations were solved for a MIP gap of 0.01. The computations were run on PCs with 4 core Intel i-5 (2.6 GHz) processor and 4 GB RAM. Each of the scenarios presented in the results were averaged over 400 iterations.

This numerical study evaluates the performance of the different schemes under different traffic conditions and on different network topologies. Three different topologies are used in this study:

- NSFNet topology (14 routers, 40 directed links)
- Ring Topology (14 routers, 28 directed links)
- 4x4 Grid Topology (16 routers, 48 directed links)

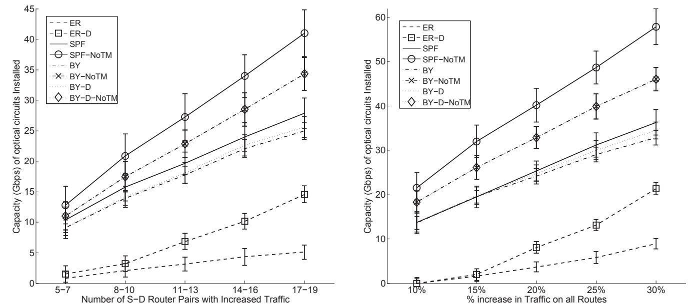
For each network, a base traffic matrix is used to generate the initial loading conditions and the link capacities are dimensioned based on this traffic matrix. Initial routing and link capacities of the network are computed using the SPF model under additional constraints, where L_{ij} is constrained to be equal to 1 if a link exists in the initial network topology, and is constrained to be 0 otherwise. For each iteration, the base traffic matrices are generated to provide different initial configurations for the network. In case of the NSFNet topology, a scaled version of the representative traffic matrix presented in [51] is used and the base topology and traffic matrix instances for each iteration are generated by scaling the independent traffic values using a uniformly random scaling factor in the range [0.9, 1.1]. For the ring and the grid topologies, initial traffic matrices are generated randomly, with traffic between every source-destination pair generated using a uniform random distribution between [0.4,1.2] Gbps. In each study, the underlying optical network is also assumed to have the same network topology. This study focuses primarily in identifying the differences in terms of the optical capacity requirement and placement for different IP routing schemes, and therefore assumes unlimited available capacity on the optical network. As a result, we do not enforce constraints on routing of circuits in the optical network presented in Section 2.5, and the parameter $HopCount_{xy}$ is evaluated as the hop count on the shortest path on the physical topology.

In the course of this study, the link utilization threshold $\alpha = 0.7$, link weights $\hat{w}_{ij} = 1$ and the number of dynamic optical circuits that can be aggregated between the same pair of routers (N) is set to 1.

3.5.1 Performance Under Different Traffic Conditions

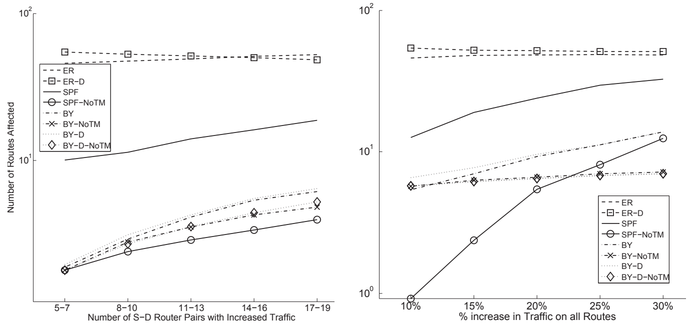
This study evaluates the performance of the different schemes for two overloading scenarios, indicative of a) traffic churns on a small number of routes in the network and b) traffic increase across all routes in the network. For each scenario, overloaded traffic matrices are generated by manipulating the base traffic matrices. In the *first scenario*, a small number of router pairs in the base traffic matrix are selected and the traffic on these routes is increased by 150% and different overloading conditions are generated by changing the number of router pairs on which traffic is increased. In the *second scenario*, traffic on all routes in increased by a fixed percentage, and different overloading conditions are generated by changing the factor by which traffic is increased on all links.

The performance in terms of the installed optical capacity and



(a) Optical circuit capacity required in Scenario 1

(b) Optical circuit capacity required in Scenario 2



(c) Number of routes affected in Scenario 1

(d) Number of routes affected in Scenario 2

Figure 3.4: Performance of different schemes on the NSFNet network for different traffic overloading scenarios: (Scenario 1) Traffic on randomly selected router pairs is increased by 150%; (Scenario 2) Traffic on all router pairs is increased by a factor as indicated. Error bars indicate 95% confidence intervals. (Confidence intervals on the results for routing changes were very small and not been shown in the figures for clarity.) [1]

the number of affected routes for the different traffic overloading scenarios is presented in Fig. 3.4. The required optical capacity for both traffic scenarios (Fig. 3.4(a) and Fig. 3.4(b)) follows the same trend, albeit with Scenario 2 requiring higher optical capacity. The results indicate that the nature of the traffic change does not significantly affect the performance of the schemes in terms of optical capacity requirements.

From these results, it is clear that the ER mechanism with complete control on the routing of individual traffic flows has the lowest optical capacity requirements. The constraints on destination-based forwarding in ER-D marginally increases the capacity demand as compared to ER, but the overall capacity demand in this scenario is still lower than the other schemes. The three schemes under known traffic conditions (SPF, BY, and BY-D) have very similar performance under both scenarios. However, it is interesting to note that the BY mechanism has a marginally lower optical capacity requirement, followed by BY-D, and then by SPF. This result is non-intuitive as the Optical Bypass based schemes are constrained to use segments on original routes, while the SPF scheme can effect large-scale routing changes by inserting new links in the network. However, the nature of SPF requires that all routes where the shortest path length is affected by the introduction of a new link be re-routed, while the BY-based schemes have better control on the selection of the traffic that is re-routed onto the dynamic optical circuits, thus leading to the lower demand on optical capacity.

Another interesting result is the performance of the schemes under unknown traffic matrix conditions (Optical Bypass under Unknown Traffic Conditions (BY-NoTM), Optical Bypass with Destination-Based Forwarding under Unknown Traffic Conditions (BY-D-NoTM) and Shortest Path First Routing under Unknown Traffic Matrix Conditions (SPF-NoTM)). Both Optical Bypass based schemes under unknown traffic matrix conditions have almost equal performance, due to the nature of the traffic expressions generated in the traffic expression set D. Here, in case of two-hop optical bypasses, all traffic that initially traverses these 2 hops is offloaded onto the optical bypass, the traffic on the optical bypass (γ_{ik}^{j}) and the resultant link load on the existing links ($LinkLoad_{ij} - \gamma_{ik}^{j}$, $LinkLoad_{jk} - \gamma_{ik}^{j}$) can be computed accurately, thereby leading both schemes to compute almost identical solutions. In the case of the SPF-NoTM mechanism, the introduction of a new link leads to possibly many routing changes, which may make it harder to find matching expressions in D for the same. As a result, the required optical capacity in the case of SPF-NoTM is significantly larger than the BY-NoTM and the BY-D-NoTM mechanisms.

The number of routes affected by the different mechanism for both traffic overloading scenarios are presented in Fig. 3.4(c) and Fig. 3.4(d). The low optical capacity requirements of the ER and ER-D schemes is a result of their capability to flexibly configure IP routing, which can be seen from the results that indicate the number of routing changes in both schemes to be an order of magnitude higher than the other mechanisms. The SPF scheme also demonstrates significantly large number of routing changes as compared to the BY and BY-D schemes, further highlighting that the SPF mechanism is not ideally suited for offloading traffic onto optical circuits in dynamic network scenarios. The large number of affected routes in the traditional schemes is the main reason behind the reluctance to use dynamic optical circuits in IP networks.

The Optical Bypass-based mechanisms under unknown traffic conditions (BY-NoTM and BY-D-NoTM) demonstrate fewer routing changes as compared to their counterparts under known traffic matrix conditions. This is indicative of the fact that under unknown traffic matrix conditions, solutions prefer to increase capacity of existing links rather than making new links, leading to higher optical capacity demands and lower routing changes. Finally, the SPF-

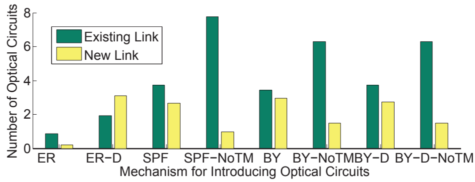


Figure 3.5: Optical circuits as new links or capacity boost for existing links when traffic of 17-19 s-d pairs in NSFNet is increased by 150% [1]

NoTM solution in Scenario 1 demonstrates the least amount of routing changes, clearly indicating that the solution prefers to increase the capacity of existing IP links rather than creating new IP adjacencies that make it difficult to find traffic expressions in D to match link load expressions. These conclusions are also supported by the result presented in Fig. 3.5, which indicates the number of optical circuits used by the different schemes to a) increase the capacity of existing IP links or b) create new IP adjacencies. The result also highlights that the BY based schemes under unknown traffic matrix conditions are better at evaluating the traffic on new links as compared to the SPF-NoTM scheme, indicated by the fact that both BY-NoTM and BY-D-NoTM schemes employ more dynamic optical circuits as new links when compared with the SPF-NoTM scheme.

Fig. 3.6 demonstrates the advantages of aggregating multiple dynamic optical circuits in terms of the installed optical capacity. As seen from the values in Table 3.1, the difference in interface granularities is significant, and as a result, matching interface size to the actual demand for optical capacity is not easy. Simulations performed for the same traffic conditions but with $N = 4$ show that for each approach, the required optical capacity is reduced when multiple circuits are allowed to aggregate, and the difference in required optical capacity is especially pronounced when the traffic matrix is

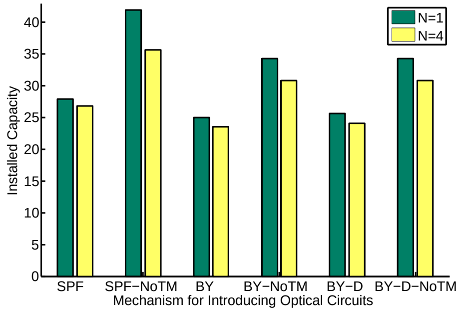


Figure 3.6: Difference in installed capacity with larger N when traffic on 17-19 s-d pairs in NSFNet is increased by 150% [1]

not known. This can be attributed to the fact that the limited traffic information forces solutions under unknown traffic matrix to use more high-capacity interfaces and as a result, the aggregation of multiple lower capacity optical circuits can better match the capacity demand, leading to a decrease of almost 5 Gbps as seen in the case of SPF-NoTM. Other studies [52] focusing on the routing of multiple dynamic optical circuits have also demonstrated similar conclusions.

Effect of Additional Constraints on SPF Performance

The formulation for SPF presented in Chapter 2 includes constraints on shortest-path selection, destination-based forwarding and the routing re-convergence constraints which are novel to the formulation. The effect of these constrains on the performance in terms of the required optical capacity is demonstrated by a study on a single network overloading scenario for five different schemes, namely

- Explicit Routing (ER)
- Explicit Routing under Destination based Forwarding (ER-D)
- Shortest Path Routing (SP)

Table 3.2: Optical capacity required under limited combination of routing and forwarding constraints [1]

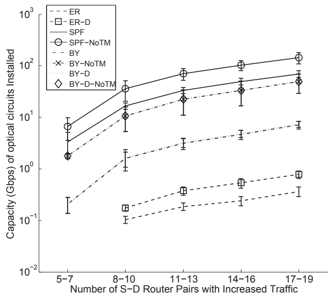
ER	ER-D	SP	SPDF	SPF
5.61	14.48	14.63	16.79	26.12

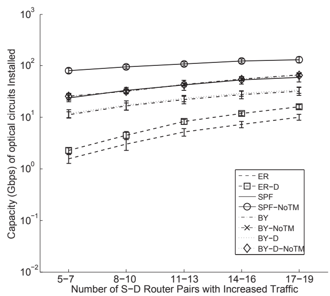
- Shortest Path Routing with Destination-based Forwarding (SPDF)
- Shortest Path First Routing (SPF)

The required optical capacity for each of these schemes is computed when traffic on 17-19 source-destination router pairs in the NSFNet topology is increased by 150%, and is presented in Table 3.2. The constraints on shortest-path routing, destination-based forwarding and routing re-convergence, when applied to the basic explicit routing formulation, have a significant impact on the required optical capacity in the network. For example, the constraint on destination-based forwarding, when applied to explicit routing (ER-D), significantly increases the required optical capacity as compared to ER, which is also the case with constraints on shortest path routing SP and ER. Similarly, the additional constraint on routing re-convergence, which is applied in SPF, leads to an increase of almost 60% in terms of the required optical capacity as compared to the SPDF scheme. These results show that ignoring these constraints when modeling SPF can lead to significant under-estimation of the required optical capacity.

3.5.2 Performance in Different Topologies

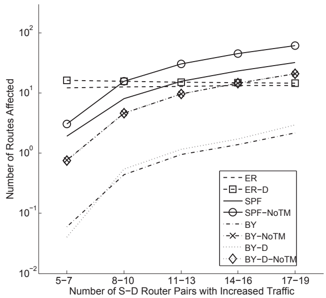
This study evaluates the performance of the different mechanisms on the 14 router ring topology and the 16 router grid topology. The performance is measured in terms of the optical capacity installed and the number of routes affected and is presented in Fig. 3.7.

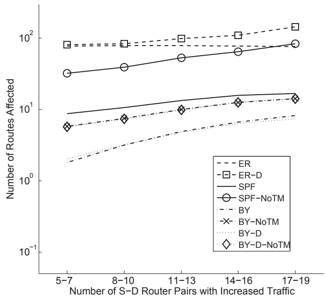




(a) Optical circuit capacity required in the ring topology

(b) Optical circuit capacity required in the mesh topology





(c) Number of routes affected in the ring topology

(d) Number of routes affected in the mesh topology

Figure 3.7: Performance of different mechanisms in the 14-node ring and the 4x4 grid (mesh) topology when overloading is achieved by increasing the traffic on a number (as indicated on the X-axis) of randomly selected source-destination pairs by 150 % [1]

In case of the ring topology, the installed optical capacity for all mechanisms increases almost linearly with the increase in the number of overloaded routes as seen in Fig. 3.7(a). Similar to the previous study, the ER and ER-D mechanisms display the lowest requirements on optical capacity followed by the BY and BY-D mechanisms. The difference between the BY and BY-D mechanisms in case of the ring topology is very small owing to the unique routing configuration of a ring network. The difference between the Explicit Routing based schemes and the Optical Bypass based schemes is much less pronounced in the ring network as compared to the NSFNet network, owing to the relatively low options for alternate routes available for the ER based mechanisms. Also similar to the previous solutions, both the Optical Bypass based mechanisms under unknown traffic conditions (BY-NoTM and BY-D-NoTM) have almost equal requirements on optical capacity which is marginally higher than the same mechanisms under known traffic matrix conditions. However, unlike in the NSFNet topology, the optical capacity required by the SPF mechanism is significantly higher than the Optical Bypass based mechanisms. This is primarily due to the fact that Optical Bypass based schemes can establish multi-hop dynamic optical circuits without significantly affecting the routing of traffic in the network, but a multi-hop circuit established in the case of the SPF mechanism affects a large number of routes, and as a result, the actual solution primarily prefers to increase the capacity of existing IP links. The performance of the SPF mechanism in these conditions is worse than even the BY-NoTM and BY-D-NoTM schemes, while the performance of the SPF-NoTM scheme is marginally higher than that of the SPF scheme as expected.

The regular nature of the ring topology leads to the overloading of consecutive links, and optimal solutions (ER and ER-D) in such a topology would employ multi-hop optical circuits. For example, in the case when traffic on 17-19 router pairs is increased by 150%, the ER and ER-D schemes only use 3 or more hop dynamic optical circuits. The next best schemes, the BY (30%) and BY-D (29%) use a smaller fraction of long (≥ 3) circuits as they are constrained by the original routing in the network, but still manage to perform better than the SPF (25 %) scheme.

These trends are also highlighted by the results for the number of affected routes as seen in Fig. 3.7(c). The number of routes affected by the BY and BY-D mechanism are the lowest, followed by the BY-NoTM and the BY-D-NoTM mechanisms, while the number of routes affected by the SPF and SPF-NoTM are significant. As compared to the NSFNet topology, the number of routes affected by the ER and ER-D mechanisms is significantly lower due to the fewer options in terms of alternate routes in a ring topology.

The overall trend in the case of the grid topology in terms of the required number of optical circuits (Fig. 3.7(b)) is very similar to the ring topology, but has some unique features. Here, the performance of the SPF and the BY-NoTM and the BY-D-NoTM schemes is very similar. This is a result of the highly regular nature of the network topology, where routing changes, especially in the case two-hop optical circuits employed as a diagonal in a single grid can be easily determined using the expressions in the set D. As a result, even though the number of affected routes in the case of BY-NoTM and BY-D-NoTM are marginally lower than SPF (Fig. 3.7(d)), the total installed capacity is very similar. The grid topology also offers a large number of alternate routes between routers, which allows the ER and ER-D schemes to flexibly change routes, as indicated by the higher number of affected routes observed in the results.

The results not only show that the SPF scheme is not well suited to regular topologies but also indicate the importance of accurately modeling the routing re-convergence constraint. Especially in the case of the grid topology, if routing re-convergence constraints are not employed, the SPF mechanism can re-route traffic onto one of the many possible alternate paths without the establishment of new dynamic optical circuits, thereby reducing the demand for additional optical capacity.

3.5.3 Effects of Dynamic Optical Circuit Decommissioning

In the studies presented till now, traffic was always increased from a stable traffic condition leading to the introduction of new optical circuits. This study analyses the capability of the proposed model to compute if dynamic optical circuits should be *decommissioned*. The objective function (2.12) allows dynamic optical circuits to be decommissioned and for the purpose of this study, the parameter $SW_{xy}^t = 0.7 \times Cost_{xy}^t$[2].

This study is performed on the NSFNet topology, and the network is significantly overloaded in the first step and each subsequent step reduces the overloading in the network. In this study, the overloading in the first step is achieved by selecting 18 random source-destination router pairs, and increasing the traffic between these pairs by 120%. In each subsequent step (steps 2-4), traffic on three of the originally selected 18 router pairs in step 1 is reset to the original traffic value. This process simulates a temporary traffic overload in step 1 which gradually reduces in the subsequent steps.

Fig. 3.8 shows the required optical capacity at each step, and clearly demonstrates the capability of the proposed formulation to switch-off optical circuits with the decrease in network load. The demands on the required optical capacity follows a similar trend to that seen in the first study, with the ER and ER-D having the lowest optical capacity requirements, followed by the BY, BY-D and the SPF solutions.

For the models under unknown traffic matrix conditions, additional traffic expressions are included in the set D to determine traffic

[2] Minor changes to the scaling factor (0.7 in this case) do not make a difference in the total outcome of the ILP.

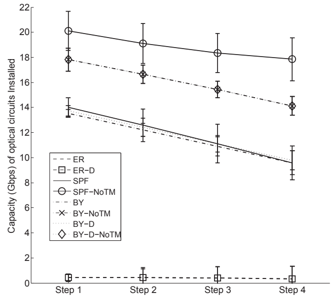


Figure 3.8: Average installed capacity at each step for switch-off in the NSFNet topology. [1]

on existing links after dynamic optical circuits are decommissioned. In the scenario, if a dynamic optical circuit is established from v_x to v_y, the decommissioning of this optical circuit will lead to an increase in traffic on other IP links in the network. To generate possible traffic expressions for the same, additional traffic expressions $LinkLoad_{ij} + LinkLoad_{xy}$ for all links from v_i to v_j, where $\psi_{ij}^{xy} = 1$ are included in the expression set D. As seen from the results, the model under this additional set of traffic expressions, especially in the case of the BY-NoTM and BY-D-NoTM can follow the trend of the corresponding solutions under known traffic matrix conditions.

A change in the link topology is likely to cause a significant change in routing, especially in the case of ER- and SPF-based routing models. However, while the SPF model does not require any additional configuration in the network, the ER and BY based models would require explicit configuration in the network. The configuration overhead associated with each of these schemes is evaluated as the number of configuration required at each step in the simulation. The number of configurations required for

- ER is computed as the difference in the routing variable r_{ij}^{sd}

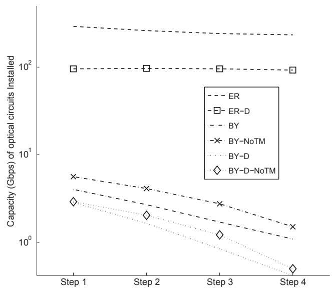


Figure 3.9: Average routing configurations at each step for switch-off in the NSFNet topology. [1]

- ER-D is computed as the difference in the forwarding variable $FT_i^d(j)$
- BY and BY-NoTM is computed as the difference in the variable f_{xy}^{sd}
- BY-D and BY-D-NoTM is computed as the difference in the variable $f_{xy}^{d} \cdot \psi_{xy}^{sd}$ [3]

The number of routing configurations required at each step are presented in Fig. 3.9. As expected, the flexibility in routing in the ER and ER-D schemes leads to a very high configuration effort which is orders of magnitude higher that that for the BY based schemes. In these schemes, the solutions under the unknown traffic matrix conditions BY-NoTM and BY-D-NoTM display lower routing configurations than their counterparts under known traffic matrix conditions, but in both cases, the total number of configurations required is less than 10 configurations per step. This result further outlines the advantages of using BY-based schemes in dynamic traffic

[3] While the number of routing configurations would be given by f_{xy}^{d}, the configuration overhead also looks as all flows affected in this case in terms of overhead on configuration of measurement and failures

Table 3.3: Computation times (seconds) in the NSFNet topology [1]

N	ER	ER-D	SPF	SPF-NoTM
1	1471.68 ± 83.52	945 ± 80.23	190.65 ± 45.92	61.35±8.53
4	-	-	270.71± 64.86	83.42±10.27
N	BY	BY-NoTM	BY-D	BY-D-NoTM
1	0.14 ± 0.01	5.27 ± 0.03	0.05 ± 0.01	4.41 ± 0.07
4	0.22 ± 0.04	7.34 ± 0.05	0.09 ± 0.01	6.82 ± 0.13

conditions, as with a relatively low management overhead, these schemes can offload traffic onto optical circuits with fewer affected routes and lower optical capacity that the traditional SPF-based solutions. The solution is of significant importance especially due to the ability to compute near-optimal solutions using only basic traffic measurements that are readily available in commercial routers.

3.5.4 Computation Complexity

Another important feature of the Optical Bypass based mechanisms is the significantly lower computational complexity, making it ideal for use in dynamic network conditions. Table 3.3 presents the measured computation times for the different schemes in the NSFNet topology.

The computation complexity of the ER and SPF schemes is dependent on the routing variable r_{ij}^{sd} (variable count $O(|V|^4)$), and in this particular model, the ability to include multiple new links further increases the computation complexity. As presented in [53] variants of the unsplitable SPF routing problem have been shown to be NP-hard in fixed network topologies, and the results clearly show that the SPF, ER and ER-D models have very high computation times, making their application unfeasible in dynamic network scenarios[4].

[4] The measured computation times showed that while the average computation

In comparison, the Optical Bypass based models have significantly lower computation complexity as the capability to re-route traffic is significantly constrained in these models. The complexity is dependent on the choice of the boolean variable f_{xy}^{sd} and f_{xy}^{d} in the case of BY and BY-D respectively, and the corresponding variable counts in both cases is of the order $O(|V|^2 \cdot H^2)$ and $O(|V| \cdot H^2)$ respectively, where H is the average hop count for routing paths in the network. Also, unlike the ER- and SPF-based schemes, the location of optical bypasses are automatically constrained by the location of overloaded IP links in the network, as the introduction of an optical bypass can only affect the traffic on IP links that it bypasses. As a result, the measured computation times for the BY and BY-D solutions are significantly lower than that of the ER- and SPF-based schemes.

An interesting factor here is that the computation times of the solutions under unknown traffic matrix conditions, especially in the case of SPF, are low even with the increase in the number of variables. This is primarily due to the fact that solutions where link load expressions for computed routing are not found in D are easily rejected due to the use of traffic bounds, thereby reducing the solution space and reducing the time complexity.

3.6 Summary

The Optical Bypass approach proposed in this thesis is a novel approach which is designed to address management challenges in IP networks associated with the introduction of new optical circuits and the consequent affect on IP routing. The proposed approach ensures that the introduction of optical circuits does not affect the routing protocols running in IP networks, and consequently affects a very small number of routes when introducing dynamic optical

time for the SPF model in case of $N = 4$ was 270 seconds, the worst case time observed was approximately 6500 seconds.

circuits.

The numerical analysis presented in the chapter compares the performance (across different topologies and traffic overloading conditions) of the Optical Bypass-based approaches with the traditional IP routing schemes presented in the previous chapter. The analysis indicates that the BY technique requires less optical capacity and has significantly lower impact on number of affected IP routes than the traditional SPF-based schemes. The computation complexity of the BY technique is also significantly lower than the SPF and ER-based techniques, making it ideal for application in dynamic network scenarios.

The combination of the BY technique and the formulation for computation under unknown traffic matrix conditions demonstrated here is especially significant. The results show that the formulation can compute near-optimal solutions for the use of optical bypasses under unknown traffic matrix conditions, making it ideal for application in real network scenarios.

Variations of the proposed optical bypass mechanism are being employed by network operators on a limited basis within their networks, and operators are also exploring the application of these solutions to other network scenarios such as cloud-bursting, scheduled backup across data centers, and even reducing the energy consumption of the network. Preliminary studies into the application of the optical bypass for reducing energy consumption in the network have been performed and are presented in [54, 55, 56].

4

Multi-layer PCE-based Service Provisioning

One of the primary challenges within packet-optical integration is the ability to dynamically provision and utilize the capacity in the two networks. Even after large-scale attempts such as the development of integrated multi-layer control plane [57] and NMS solutions [58], most carriers typically have independent management infrastructure in place for both networks, and any coordination between them is typically carried out by human operators. While some of these issues can be addressed by upcoming SDN frameworks, the implementation and integration of specialized management operations such as path computation, which are dependent on the network technology (and in some cases the vendor) poses a significant challenge.

To this end, standardized third-party management sub-systems are being developed which provide targeted vendor-independent capability to perform specific operations in a network. The vendor and technology independent interfaces to these subsystems can be employed to facilitate automated coordination across multiple network layers. The Path Computation Element (PCE) is the prime example of such subsystems, and standards for the PCE have been proposed to support path computation and provisioning in multiple technologies and across domain and multi-layer boundaries. The standardized interfaces to the PCE allow the easy integration of optimized third-party PCE implementations to be integrated with existing NMSs, control planes and even upcoming SDN frameworks.

The *first open-source PCE*[3, 59] was developed as a part of this thesis work. An open-source PCE implementation is an important component for developing and evaluating proposals for PCE standardization, and this chapter highlights some of the design and implementation features of the open-source PCE that allow the implementation to be extended and customized to adapt to various network and system scenarios. The chapter also presents an overview of the first experimental demonstration of *PCE-based multi-layer path computation and provisioning* performed with Telefonica I+D, Spain, that identifies challenges and provides insights on standardization gaps that must be addressed for multi-layer service provisioning over the PCE architecture.

4.1 Supporting Publications

1. O. Gonzalez de Dios, V. Lopez, M. Cuaresma, F. Munoz, M. Chamania, A. Jukan,“Coordinated Computation and Setup of Multi-layer Paths via Inter-layer PCE Communication: Standards, Interoperability and Deployment,” to appear in **IEEE Communications Magazine**, 2013 (Telecommunications Standards Series).

2. M. Chamania, M. Drogon, A. Jukan, “An Open-Source Path Computation Element (PCE) Emulator: Design, Implementation and Performance,” **IEEE/OSA Journal of Lightwave Technology**, vol.30, no.4, pp. 414-426, 2012.

3. S. Martinez, V. Lopez, M. Chamania, O. Gonzalez de Dios, A. Jukan, J. P. Fernandez-Palacios, “Assessing the Performance of Multi-Layer Path Computation Algorithms for different PCE Architectures,” **OSA Optical Fiber Communication Con-**

ference (OFC) / National Fiber Optic Engineers Conference (NFOEC), 2013.

4. M. Chamania, O. Gonzalez de Dios, V. Lopez, M. Cuaresma, M. Drogon, A. Jukan, X. Masip-Bruin, M. Yannuzzi, "Coordinated Computation of Multi-layer Paths via Inter-layer PCE Communication: Standards, Interoperability and Deployment," **IEEE International Conference on Communications (ICC)**, 2012.

5. **POST-DEADLINE PAPER** M. Chamania, M. Drogon, A. Jukan, "Lessons Learned From Implementing a Path Computation Element (PCE) Emulator," **OSA Optical Fiber Communication Conference (OFC) / National Fiber Optic Engineers Conference (NFOEC)**, 2011.

4.2 Overview of the IETF PCE Architecture

The IETF PCE [60] was envisioned as a third-party management entity to perform QoS constrained path computation. The PCE is a centralized server, which serves requests for path computations using topology information stored in its Traffic Engineering Database (TED). The TED contains topology and traffic engineering information, as well as technology-specific parameters, which are necessary for path computation in specialized technologies such as Wavelength Division Multiplexing (WDM). The request for path computation is generated by a Path Computation Client (PCC) which can be a network element or a NMS, and communication between the PCC and PCE uses the Path Computation Element Communication Protocol (PCEP) [61].

The PCEP protocol allows a PCE to request paths from another PCE, thereby providing capability to extend constrained path computation in multi-domain [62] and multi-layer [63] network scenarios with minimal topology information exchange.

The separation of path computation capability from the network equipment or the NMS is very attractive for network operators. In a legacy network, deployment of one or more path computation algorithms would require software updates for the network devices or the NMSs which is costly and time-consuming. On the other hand, an update of path computation algorithms in the PCE is relatively simple and cheap. The use of the PCE also helps network operators deploy and use the same algorithms in multi-vendor networks, which is non-trivial in legacy systems. The capability to use the same interface to third-party systems for path computation in mutli-vendor, multi-technology network environments also makes the PCE an attractive proposition for upcoming SDN frameworks.

4.3 Open Source PCE

The primary objective of the PCE approach was to allow network operators to easily deploy new path computation algorithms in multi-vendor settings. However, most implementations of the PCE are vendor-specific which use proprietary extensions, especially for the management and update of the TED. Third-party commercial PCE solutions are also available, but provide limited capability for customization, which makes integration in multi-vendor networks challenging. There have been other open-source solutions proposed for path computation such as the OSCARS-IDC [64], but these solutions do not employ the PCE standards, which is not desirable in carrier networks.

The PCE implementations should also have the capability to keep up with emergence of new transport network technologies such as flex-grid [7] and OTN [8] which in turn requires updates to the existing standards. In terms of additional features such as security, topology update and policy enforcement, the PCE standards do not specify any specific mechanism, and instead allow for a number of possible standards to be incorporated into the PCE framework.

Due to these factors, it is necessary to have a open and extensible implementation of the PCE that allows developers and network operators to build and easily integrate extensions for various aspects of the PCE including (but not restricted to) new path computation algorithms. The open-source PCE presented in this thesis was designed to address these issues, and three specific design requirements were considered critical to is usability, namely,

- *Modular design*, which identifies macro operations in the PCE architecture, thereby allowing developers to independently update or replace the implementation for these operations with ease.

- *Extensibility*, which in the context of the implementation implies the ease of introduction of new modules (such as policy enforcement functions) and the generic nature of inter-module communication in order to support extensions to the message exchange between modules without significant implementation effort.

- *System adaptability*, which implies the ability to tweak the performance of individual modules to best suit the (unique) needs of the network in place. For example, a PCE for a WDM network is likely to receive very few requests but must perform complex computation, while an MPLS PCE will probably observe a higher connection load with relatively simple path computation.

The architecture of the open-source PCE is designed based on these features and some of the design highlights and novel implementation features are presented next.

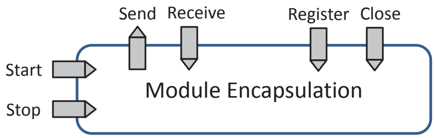


Figure 4.1: The *Module* interface indicating the interfaces exposed by all modules [3]

4.3.1 Architecture and Implementation Specifications

The design for the open-source PCE identifies components or *modules* that are likely to be customized to specific vendor/network scenarios. The choice of these modules is essential as it allows developers to update or replace a specific component, while re-using the other existing components of the open-source implementation. In order to ensure that there is minimal overhead in adding, removing or replacing modules, each module is encapsulated by a standard interface as shown in Fig. 4.1.

The primary modules identified by this architecture include the

1. **Network Module**, which is responsible for managing communication with remote PCE peers.

2. **Session Management Module**, which manages PCE sessions inside a PCE peer.

3. **Computation Module**, which is responsible for serving path computation requests.

4. **Client Module**, which provides the integration of the PCE client with the associated NMS or network device, allowing them to initiate PCEP sessions, generate path computation requests and processing path computation responses.

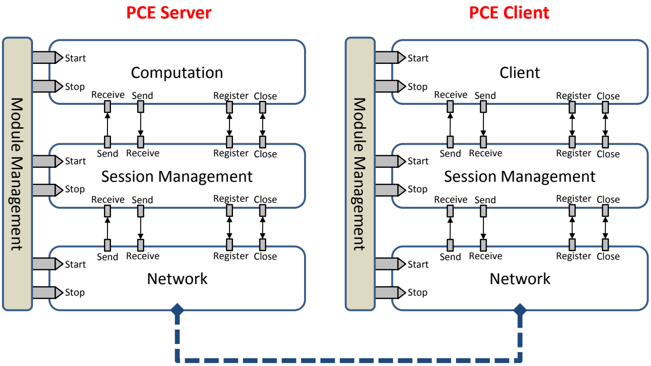


Figure 4.2: *Modular* architecture of the PCE and the PCC indicating the interactions between the different modules [3]

The interaction of these modules inside the PCE and PCC is presented in Fig. 4.2. Each module in the implementation is a separate process, and the *Module Management* process is responsible for initializing and managing the life-cycle of each of these modules. The *Start* and *Stop* interfaces on each module allow the module management process to gracefully manage the start-up and shutdown of the PCE.

The module management also plays a part in inter-module communication. The references to each of the modules in a particular implementation are present with the module management process, and a module attempting to send a message to another module requests a reference to the remote module from the module management process before sending a request to the same. For example, in the interactions shown in Fig. 4.2, the network module interacts directly with the session management module, but in the implementation, the Network module only has a reference to the module management process, and requests the same for a reference to the session management process before sending a message. This particular feature is significant as it allows for load balancing (e.g. multiple computation modules running to serve requests) and for hot-plugging of new modules, if required. In the open-source PCE, inter-module communication is implemented as a Java function call, but can be replaced by other possible solutions such as web services with relative ease, allowing the implementation of the PCE to be distributed across multiple PCs as well as facilitating the use of different programming platforms for individual modules.

Module Encapsulation and Inter-Module Communication

As specified before, each module is an independent process, which is launched by the module management process, and the life-cycle of a module is managed using the *Start* and *Stop* interfaces.

Inter-module communication within the open-source PCE is related to processing messages belonging to a particular PCEP session. Different modules can initiate a PCEP session in the PCE - the request to start a PCEP session in the PCC originates at the client module, while the PCE identifies the request for a new session when a new connection request arrives at the network module. The *Register* interface on each module allows modules to communicate the arrival of a new PCEP session request to the other modules in the implementation, and the *Close* interface allows the modules to communicate the termination of a specific PCEP session.

Within one PCE instance, it is necessary to have an ID for a PCEP session that can be easily generated in both the client and the network modules. The implementation uses the combination of the remote PCE peer IP and TCP port as the internal session identifier. The standards specify that there should only be a single session established between any two PCE peers, and this can be additionally constrained on the implementation if required.

The *Send* and *Receive* interfaces exposed by the modules support asynchronous communication of PCEP messages between the modules. The *Send* interface in a module allows processes within a module to send a PCEP message to another module using the *Receive* interface exposed by the module. The interfaces also specify the internal session ID to identify which the PCEP session a message is associated with and the remote module i.e. the module to which the message is being sent to in case of the *Send* interface and the module from which the message was sent in case of the *Receive* interface.

The use of the PCEP messages is sufficient for operation of each of the modules presented in Fig. 4.2, and the object-oriented implementation of the messages themselves allows developers to easily incorporate extensions to the PCEP messages without any effect on inter-module communication. The implementation additionally includes the address of the remote peer (acting as the ID) in the PCEP message to enable the mapping of a PCEP message to a particular PCEP session.

Object-Oriented Representation of the Path Computation Element Communication Protocol (PCEP)

The PCEP is a critical part of the PCE framework, and any updates or extension to the same usually involve updates to the protocol itself. A PCEP message is a single message that is exchanged between any two PCE peers and consists of a 32 bit *Header* which indicates the protocol version, the type of message as well as the message length. Data within the PCE message is a collection of objects, each of which themselves are defined using a header and a value. The value of a PCEP object can be primitive data type or a collection of PCEP Objects, thus enabling complex object definitions.

Specific messages in the PCEP protocol are associated with a list of required and optional objects that are used to verify if a message is correct. As a result, the content of a *PCEP Message* is encapsulated

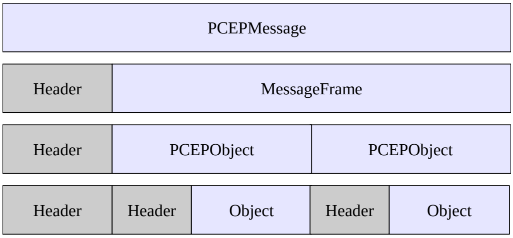


Figure 4.3: Object hierarchy of the PCEP message implementation [3]

in a *Message Frame* and implementations of the *Message Frame* for specific message types implement the logic for checking if the objects inside the message conform with the requirements of the PCEP standards.

In the implementation, primitive interfaces for the *PCEPMessage* and *PCEPObject* are defined and a *factory* pattern is used to define a single location where messages and objects are generated. The use of the interface allows developers to easily generate new objects, and the factory pattern allows the easy integration of these objects into existing messages. Each component (header, PCEPObject and PCEPMessage) of the PCEP implementation also contains a serialize and de-serialize method to allow the conversion of objects from bit-streams to primitive data types and vice-versa, implying that the conversion of messages in the network module for transmission is not affected by the implementation of new objects.

Network Module

The Network module is responsible for managing the communication channels or sockets between PCE peers. The network module establishes and maintains TCP connections with remote peers, and is used to exchange PCEP messages with remote peers. The Network mod-

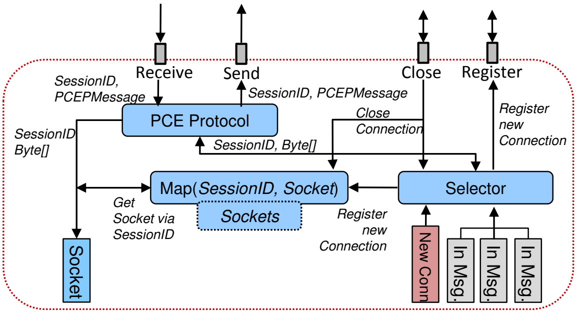


Figure 4.4: The implementation of the Network Module with Asynchronous Network I/O [3]

ule implemented in the open-source PCE does not implement any form of admission control, but can be easily extended to implement the same to manage the load on a PCE server.

The Network module implementation, shown in Fig. 4.4, uses the Java NIO library [65] to facilitate asynchronous read/write operations. Asynchronous I/O implementation for network communication allows the network module to serve a large number of PCEP sessions with relatively low processing overhead. In this implementation, *read* listeners on established connection sockets are registered with the *Selector* process. In case any data arrives on a socket, the read listener generates an interrupt which is sent to the Selector thread, which processes the incoming data and uses the PCEP protocol library to convert the bit-stream into a PCEP message which is then sent to the Session Management module. The *receive* interface in the network module receives PCEP messages that must be sent to the remote peers. The messages are converted to bit-streams using the PCEP protocol library, and the session ID provided with the incoming PCEP message is used to extract the socket associated with the session. The socket is extracted from the Map data structure within the network module and is used to send the message to the remote peer.

In case of the PCC, the request to initiate the session is generated by the client module, and is send to the Network module over the *Register* interface. The Network module first checks if an active session with the remote peer is already available, and if not, attempts to create a connection using the IP address and TCP port parameters provided in the session ID. If a connection is established successfully, the Network module registers the *read* listener for the socket with the Selector and creates a Map entry <*session ID, socket*> in the Network module which is used to identify the socket associated with a particular session ID during write operations.

In case of the PCE, an additional *accept* listener is registered with the Selector, which listens for new connections on a pre-defined port (default 4189, as defined in the PCE specifications) and generates an interrupt in case a new connection request arrives. The selector thread extracts the remote IP and port number from the incoming connection request and initiates a session registration process by invoking the *Register* interface on the Session Management module. Once a connection is registered, the Network module registers the corresponding socket with the Selector process and performs subsequent read/write operations as described above.

When the *Close* interface is invoked, the Network module terminates the TCP socket and removes the corresponding entries from the internal map and the selector process.

Session Management Module

The Session Management module implements the logic for managing PCEP sessions inside the PCE. A PCEP session is described by the state machine shown in Fig. 4.5. The state machine is initialized in the *Idle* state, and when a TCP connection request is initiated, it transitions to the *TCP Pending* state. In case a TCP connection

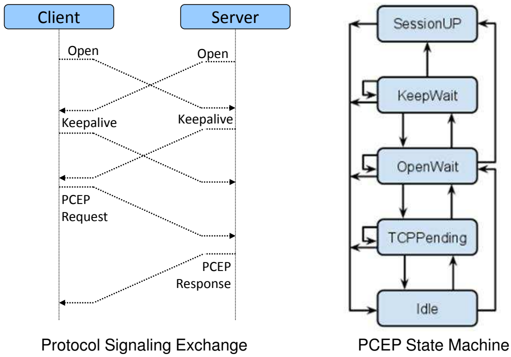


Figure 4.5: The PCEP state machine and associated PCEP message exchange between PCE peers [3]

cannot be initialized, it transitions back to the *Idle* state, and if a session is successfully created, it sends a PCEP *Open* message to the remote peer and transitions the *Open Wait* state. The *Open* message is used to indicate that the TCP session was established and recognized successfully by both peers, and is also used to negotiate session parameters such as timeout values between the peers. Upon receiving an *Open* message from the remote peer, the state machine can either accept the parameters sent by the remote peer, send a *Keepalive* message and transition to the *Keep Wait* state, or send another *Open* message with updated parameters and remain in the *Open Wait* state. If a PCE in the *Keep Wait* state receives a *Keepalive* message from the remote peer, the session establishment is complete and the state machine transitions to the *Session Up* state, after which PCEP request/response messages can be exchanged between peers.

In the lifetime of a session, messages must be exchanged between the peers at regular intervals to indicate that the remote peer is still active. The Session management module is also responsible to man-

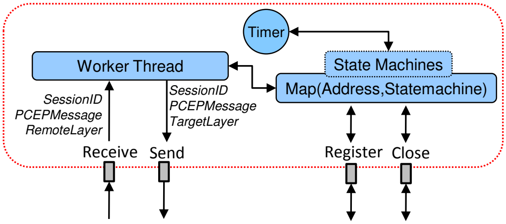


Figure 4.6: Implementation of the Session Management Module [3]

age the message timers, and to generate periodic *keepalive* messages in case no other messages are exchanged between the peers for a specific period of time. Therefore, all messages originating from or going to remote PCE peers are sent first to the Session management module in order to maintain state about the last message exchange between the peers.

The implementation of the Session Management module is presented in Fig. 4.6. As indicated before, state machine transitions occur upon the arrival of a new message or in case a timeout is observed. As a result, the state machine is implemented as a single object, and a Map structure within the Session management module keeps a record of the various active state machines and the corresponding internal session IDs. A pool of worker threads is assigned to process incoming requests from the *Receive* interface, and upon receiving a message, a worker thread uses the session ID associated with the incoming message to extract the corresponding state machine from the Map structure and process the message. The state machine determines the transitions (if any) that need to be made within the state machine, and if the incoming message (such as a PCEP request or response message) must be routed to other modules. The state machine object also generates PCEP messages such as the *Open* and *Keepalive* messages that are necessary for the ses-

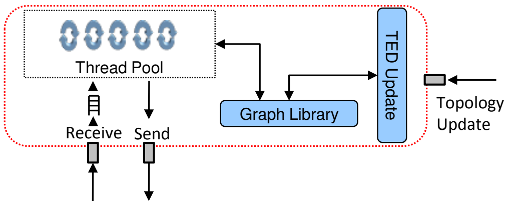


Figure 4.7: Implementation of the Computation Module [3]

sion establishment with remote peers.

In order to manage timeouts, a single timer thread is used to register events from each state machine, and events associated with a timeout are registered in this thread. Upon the arrival of a message, the corresponding state machine can reset the associated timeout events in the timer, and in case a message does not arrive in time, the timer thread invokes the corresponding operations (e.g. termination of a session/sending of a Keepalive message, etc.).

A standard interface for state machine objects is defined in our implementation, which allows developer to easily extend the current state machine implementation without significant change to the implementation of the Session Management module itself.

Computation Module

The Computation module is responsible for serving path computation requests in the PCE server. Path computation requests in the PCEP protocol are defined as a request between two endpoints along with a set of constraints. Based on the constraints defined, the Computation module can choose an appropriate algorithm to compute the request. The PCC can also indicate the choice of algorithm to be used to serve a path computation request using the *Objective function* parameter in the path computation request message.

The implementation of the Computation module is shown in Fig.

4.7. A worker thread pool is used to process incoming path computation request messages. The threads make use of the network topology information represented as a graph library object, and perform path computation using the same. The TED update mechanism is implemented as a separate process which listens for TED updates from external processes, and different mechanisms for the same can be implemented to update the topology in different network deployments.

The choice and implementation of the path computation algorithms is linked to the graph library used for representing the TED, and changes to the same leads to significant changes to the implementation of the computation module. However, the use of a graph library built into the implementation of the computation module significantly reduces processing overhead for path computation requests.

4.3.2 PCE Performance Evaluation

From the perspective of a network designer, typical performance metrics to evaluate the scalability of a PCE server consist of parameters such as maximum load served (in requests/second) and the average duration of a PCEP session required to serve a request. In this study, a single open-source PCE server was required to serve requests generated by PCE clients running on 5 different desktop PCs. All PCs used in the study were a DELL OptiPlex 760 PCs with an Intel Core2 Quad CPU Q950 (2.83 GHz) and 2 GB dual-channel DDR2-SDRAM.

The first study evaluates the capability of the PCE to serve a requests arriving with high frequency. In this test, at each *client* PC, multiple instances of a PCC application were launched that initiated path computation requests with inter-arrival time between the requests given by a Poisson distribution with a mean value of 1 second. The total load on the PCE server was varied by launching

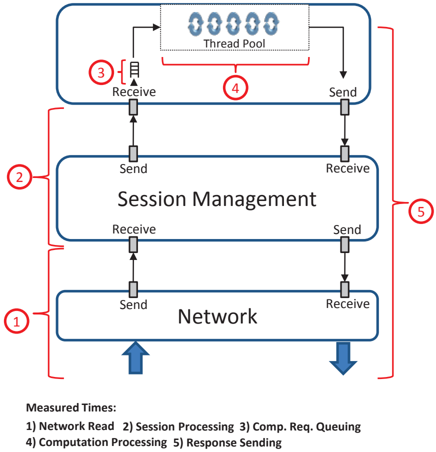


Figure 4.8: Indication of the different processing time measurements made for the PCE scalability tests [3]

multiple instances of the PCE client on the same PC.

The PCE server was required to compute the shortest (unconstrained) path between randomly selected endpoints (at the PCC) on the Atlanta network topology (15 nodes, 22 bi-directional links) [66]. The performance was computed by measuring processing times between different operations as indicated in Fig. 4.8, which include the processing of incoming packets in the network module and session management module, the queuing time in the receive queue of the computation module, the actual path computation time, and the time to process and send the PCEP path computation response message to the remote peer. The times, measured in nanoseconds are presented in Table 4.1. It can be seen from the table that the average processing times for each of the operation indicated are very low, and do not vary significantly with increase in network load. The measured variance in the processing time is relatively high and is pri-

Load		Nw. read	Session	Req. Wait	Comp.	Resp.
1500	avg	19329	46535	20254	56021	122379
	std	319281	499101	55901	369039	817393
2500	avg	20575	49515	21256	58345	123462
	std	330465	217162	60618	528935	587564
4500	avg	20796	41147	21714	60344	123919
	std	261759	180293	407260	332964	90200

Table 4.1: Time measurements (in nanoseconds) for different processes inside modules in the PCE server (shown in Fig. 4.8). Load indicates number of PCE client instances with each instance generating requests using a Poisson distribution (average inter-arrival time = 1 sec) [3]

marily caused due to the scheduling of multiple processes (threads) running within the PCE. However, the results still indicate that the PCE server, especially the network and session management module, can scale easily to serve a large number of requests, subject to the performance of the computation module.

The performance measured within the computation module is dependent on multiple factors such as TED updates, network topology size, complexity of the computation algorithm used and the number of threads used to process path computation requests. The study compares the performance of different path computation algorithms on 4 topologies, including the Atlanta (15 nodes, 22 bi-directional links) and TA2 (65 nodes, 108 bi-directional links) from [66] as well as two Internet like network topologies with (120 nodes, 237 links) and (200 nodes,397 links) generated using the BRITE random topology generator [67]. Three path computation algorithms, namely the shortest path ($O(V \cdot ln(E))$), the maximum bandwidth shortest path [68] ($O(2V \cdot ln(E))$) and the optimal link disjoint path [69] ($O(V^2)$) algorithm are used for evaluating the performance of the computa-

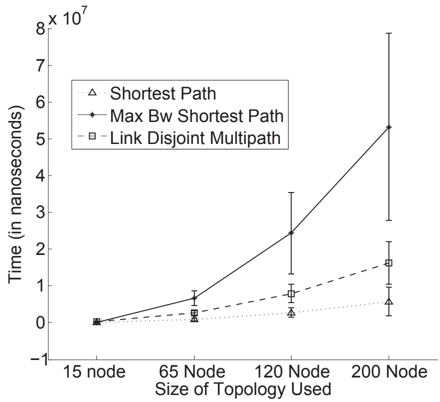


Figure 4.9: Measured processing time for path computation for different algorithms and topology sized [3]

tion module. The measured computation times are presented in Fig. 4.9, and as expected, the path computation times increase with increase in the computation complexity as well as the topology size, which in turn affects the overall performance of the PCE.

The large variation observed for the maximum-bandwidth shortest-path computation algorithm is due to the choice of random topology endpoints and the significantly larger search-space for path computation in the algorithm as compared to the shortest path and link-disjoint path computation algorithms. In such scenarios, the queuing time for requests in the computation module can be significant, and a possible solution can be to vary the number of threads used to process the requests.

The study evaluates the average queuing time and the computation time for serving path computation requests on the 200 node network topology using the maximum bandwidth shortest path computation algorithm. Path computation requests are generated using

Thread Pool Size		Queuing	Computation	Total
1 Thread	avg	5184762	50269354	55454116
	std	15121718	32560269	35479705
5 Threads	avg	226103	53045413	53271516
	std	1907186	35562894	35639152
10 Threads	avg	68466	54478004	51946471
	std	413915	32581812	32605523

Table 4.2: Measured processing times (in nanoseconds) in the request queue and for path computation when varying thread pool size for the maximum bandwidth shortest path algorithm applied to the 200 node network topology. Requests are generated using a Poisson process with the mean inter-arrival time = 400 ms. [3]

multiple PCC instances, with each instance generating requests as a Poisson process with average inter-arrival time of 400 ms.

The measurement for the average computation and queuing times for requests are performed for different sizes of the thread pool and are presented in Table 4.2. The results show that an increase in the thread pool size can significantly decrease the queuing times for the requests, while the processing times are marginally increased due to the scheduling overhead in the CPUs. However, the payoff from increasing the number of threads in the thread-pool is likely to taper off if the thread-pool size is very large, and therefore should be designed and managed carefully.

Finally, the study evaluates the effect of TED update frequency on the performance of the PCE. The management of TED updates is an implementation decision: A path computation request being processed while a TED update is received can either send a response based on the old topology, or can restart path computation based on the latest topology information. The open-source PCE implementation uses the latter and the result presented demonstrates the effect of varying TED update frequency on the residence time in the computation module (path computation time + queuing time). For this study, the 200-node network topology was used as the base network topology, and the link-disjoint path computation was performed for requests arriving as a Poisson process with mean inter-arrival time of 400 ms. TED updates were performed at fixed time intervals, with the interval varying from 0.5 to 2 seconds. The measured processing times, presented in Fig. 4.10 show that an increase in the frequency of the TED updates can increase the total path computation time. In case a TED update is received during an active path computation process, the running process is interrupted and path computation is restarted based on the updated TED. As a result, the path computation times measured in the case of a single thread are affected adversely as a restart of path computation not only increases the path computation time for the interrupted request but also affects the queuing time of the other requests. In case of a larger thread-pool size, more path computation processes are interrupted, but the average queuing time is not affected as adversely and hence the increase in the total time is smaller.

4.3.3 Applications and Extensions of the Open-Source PCE

The open-source PCE implementation is hosted on [70], and the current implementation has been extended to support multi-domain path computation in a hierarchical PCE configuration as well as multi-layer path computation which has been reported in [71, 4] and are described in the next section. Multi-path extensions have also been proposed and implemented on the open-source PCE [72].

The open-source PCE has also been used in two European research projects. The ONE project [73] is currently using the open-source PCE to perform single and multi-layer path computation. The GEYSERS project [74] has also proposed extensions for the

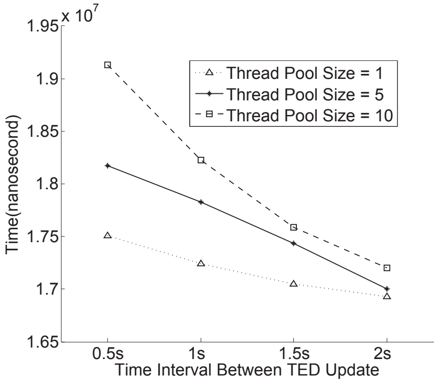


Figure 4.10: Effect of TED update frequency on processing time in the computation module [3]

PCE to support computation of joint network and infrastructure resource requests, and the proposed extensions have been implemented and developed in the open-source PCE.

Finally, the PACE project (EU-funded, to begin in end of 2013) will address all the definitions, requirements and specifications related to our open-source PCE implementation, and will also cover aspects of standardization requirements, reference models, performance metrics and interface descriptions of open-source platforms and modules as one of its unique goals.

4.4 Multi-Layer Path Computation with PCE

While the PCE can perform path computation under a variety of configurations in single and multi-domain network scenarios, its capability to perform multi-layer path computation is especially important in the current carrier network scenario for packet-optical network integration. As mentioned before, carrier networks consist of an IP/MPLS network used to provision user services such as Virtual Private Networks (VPN) and leased lines along with best-effort Internet traffic, and a Wavelength Switched Optical Network (WSON) network for providing high-capacity inter-connectivity between IP/MPLS routers. With the increase in demand for services with assured QoS, providers are looking for solutions to dynamically introduce capacity in the IP/MPLS network from the WSON network, which will lead to better resource utilization across multiple layers.

Numerous mechanisms have been proposed for the use of the PCE for multi-layer path computation and are outlined in this chapter. This chapter presents the work in [71, 4], which identifies the proposal best suited for application in current carrier network infrastructures, and presents an experimental setup that can compute and provision multi-layer paths in an IP/MPLS over WSON network. This activity was jointly carried out with partners from Telefonica I+D, Madrid, who are also actively looking to develop and employ PCE-based solutions in their networks. The activity highlights some of the challenges and *standardization gaps* that must be addressed in order to employ multi-layer provisioning in production network environments.

4.4.1 Proposals for Multi-Layer Path Computation

Multiple possible configurations for the use of the PCE for multi-layer path computation have been outlined in [63]. The proposals can be classified in a broad fashion into two general approaches, namely the *integrated* and the *coordinated* approaches.

In the integrated approach, a single PCE has information about the complete multi-layer network topology and can perform multi-layer path computation. The solution allows for the computation of optimal multi-layer paths and therefore has obvious benefits in terms of path computation [75]. However, in a real network scenario, populating and managing a common TED across multiple layers presents significant challenges. The technological and organizational separation seen between the IP/MPLS and the WSON network layers in the operators ecosystem is not compatible with the integrated PCE approach. Also, with different vendors (and consequently non-standard TED update mechanisms) used in different network layers, management of a single common TED is non-trivial.

The coordinated approach proposes the use of an individual PCE for each network layer, which is better suited for current network operator ecosystems. Solutions under the umbrella of the coordinated approach propose different mechanisms for interaction between the PCEs of the IP/MPLS and WSON network, and the complexity of the solution increases with increasing communication between the PCEs. In one proposal, the MPLS PCE only suggests possible end-points for the establishment of an optical circuit, without consultation with the WSON PCE, and the WSON PCE is requested for a path between these end-points during the provisioning process in the MPLS network. This process does not entail any communication between the PCEs in the different network layers, but is not optimal from the point of path computation as there may be multiple possible candidates for the establishment of optical circuits and the MPLS PCE chooses a solution without information about the cost associated or the availability of the optical circuit requested in the WSON network.

This solution presented in this chapter overcomes this challenge by supporting inter-PCE communication for multi-layer path computation, where the MPLS and the WSON PCE interact with each other to compute multi-layer paths. For provisioning the connections, the Virtual Network Topology Manager (VNTM) is introduced which can create optical circuits. The VNTM is an architectural concept that can be used for a myriad of purposes, including the provisioning of optical circuits as proposed by the PCE during path computation, or even suggesting the introduction of new optical circuits automatically for network engineering in the MPLS network [63, 57]. However, no standard solution for the VNTM has been proposed to date[1].

The increased cooperation between PCEs also comes at the cost of additional delay, which has also been reported in [76], but is an acceptable trade-off when compared with the potential gains in terms of the quality and consequently cost of the computed path.

An overview of the components and the associated signaling for the solution presented in this chapter is shown in Fig. 4.11. The setup consists of four major components, namely

1. IP Network Control and Management (NCM)

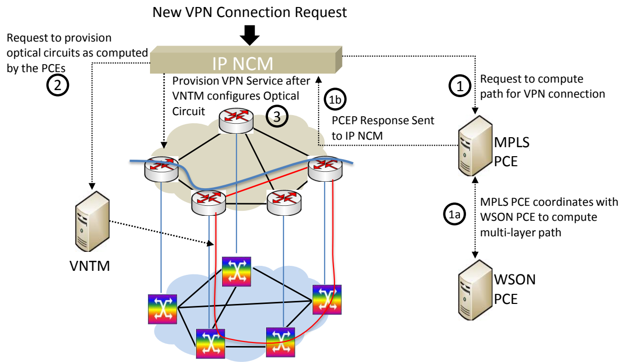
2. VNTM

3. MPLS PCE

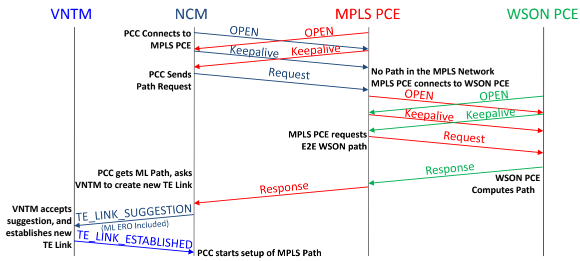
4. WSON PCE

The IP NCM is responsible for orchestrating interactions between the PCEs and the VNTM and for provisioning the service in the MPLS network. In order to setup a connection, the IP NCM requests the MPLS PCE to compute a path in the MPLS network. To send a path computation request, the IP NCM establishes a PCEP session with the MPLS PCE, which involves the exchange of *Open* and *Keepalive* messages as shown in Fig. 4.11(b), after which a path computation request is sent to the MPLS PCE. When a request arrives at the MPLS PCE, it first attempts to compute a path in the MPLS network, and if not found, selects candidate endpoints for the establishment of an optical circuit that can be used to serve this request. The MPLS PCE then establishes a PCEP session with the

[1] The ONE adapter architecture presented in the next chapter has been demonstrated to support the basic functionality of the VNTM



(a) Multi-layer Path Computation and Provisioning with PCE and VNTM



(b) Signaling interaction between various components in the presented PCE-VNTM based multi-layer provisioning solution

Figure 4.11: Automated Framework for Multi-layer connection provisioning [4]

WSON PCE, and requests the WSON PCE to compute paths between these candidate endpoints and uses the computed path information to compute multi-layer paths, with the quality of the computed solution depending on the algorithm used to determine the candidate endpoints. The computed multi-layer path is sent to the IP NCM, which forwards the computed path to the VNTM to provision the required optical circuits. As indicated before, there are currently no standards available for the interface to the VNTM and in this setup, two new PCEP messages: the *TE_LINK_SUGGESTION* and the *TE_LINK_ESTABLISHED* are proposed for communication between the IP NCM and the VNTM. Once the optical circuits are established, the NCM goes on to provision the service in the MPLS network.

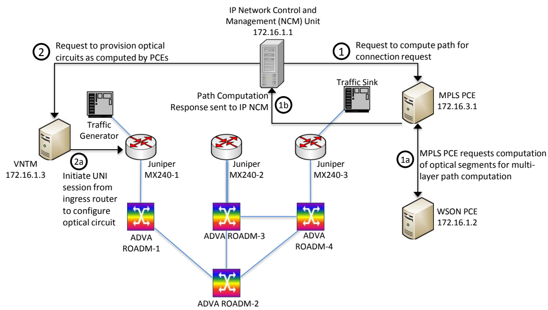
4.4.2 Experimental Validation

Testbed Setup

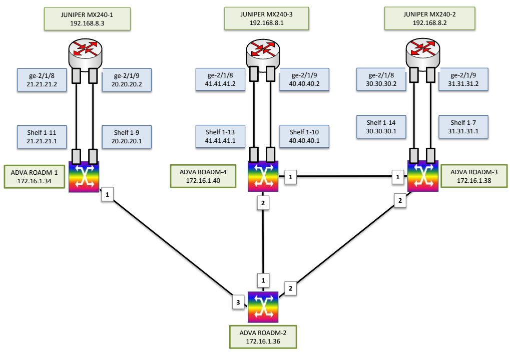
The multi-layer path computation and provisioning setup described in the previous section was developed and demonstrated on the testbed presented in Fig. 4.12. The MPLS PCE was deployed at TU Braunschweig premises in Germany and used the open-source PCE implementation, while other components in the testbed were preset at Telefonica I+D premises in Madrid, Spain. The IP NCM, VNTM and the WSON PCE were deployed on different virtual machines on a single server at Telefonica, and had negligible signaling delay between them.

The WSON PCE was a proprietary PCE implementation developed by Telefonica. The PCEs have been built on the IETF standards to ensure interoperability while the IP NCM and the VNTM used are in-house implementations for provisioning connections in the MPLS and the WSON network respectively.

The IP network consists of three Juniper MX-240 routers, and



(a) Various components used in the testbed



(b) Device and interface addressing used in the testbed

Figure 4.12: The multi-layer testbed used to demonstrate multi-layer path provisioning [4]

the WSON network consists of four ADVA ROADMs. The physical inter-connectivity and the device/interface addressing is shown in Fig. 4.12(b). As can be seen from the addressing, the control planes for the two networks are separated by ensuring unique IP subnets for the MPLS and the ROADM devices. The control plane addresses of the routers, ROADMS, IP interfaces and the client interfaces on the ROADMS are unique IP addresses, while the server (network) interfaces on the ROADMS use un-numbered interface addressing. The testbed also contains a traffic generator and sink, and successful transmission of traffic between them is used to indicate the setup of the service.

The TEDs for the MPLS PCE and the WSON PCE are updated using OSPF link state advertisements, and the MPLS PCE contains additional information about the ROADMS connected to specific routers. Initially, there are no IP links in the network, and inter-connection between routers is established by provisioning optical circuits in the WSON network. In order to provision optical circuits, the VNTM uses the User-Network Interface (UNI) interface available on the Juniper routers to establish an optical circuit between two IP endpoints.

The primary goal of this experiment is to demonstrate the capability of the setup to compute and provision multi-layer paths, and highlight challenges in terms of standardization. Therefore, the setup uses a basic path computation algorithm: The MPLS PCE attempts to compute the path in the MPLS network, and in case the MPLS PCE cannot find a path in the MPLS network, it requests an end-to-end optical circuit in the WSON network. If a path is found in the WSON network, a multi-layer path is returned as the computed path and if a path is not found in the WSON network, the path computation request fails.

Path Computation Process

The signaling exchange between the IP NCM, the MPLS PCE and the WSON PCE for computing multi-layer paths is captured using Wireshark and is presented in Fig. 4.13. Here, the IP NCM (172.16.1.1) initiates a PCEP session with the MPLS PCE (172.16.1.3) and requests a path from the router MX240-1 (192.168.8.3) to MX240-3 (192.168.8.1). In this scenario, the MPLS PCE is unable to compute a path in the MPLS network and initiates a PCEP session with the WSON PCE (172.16.1.2) and sends a path computation request to the same. For generating the request, the MPLS PCE needs to convert the MPLS endpoints to the corresponding WSON end-points, and as shown in the figure, the MPLS PCE requests the WSON PCE for a path from ROADM-1 (172.16.1.34) to ROADM-3 (172.16.1.40). The WSON PCE computes the path in the WSON network and sends a path computation response to the MPLS PCE. The Explicit Route Object (ERO) is used to define the path in the WSON network in the PCEP response, which consists of a sequence network interfaces (defined by unnumbered interface addresses) and the IP address of the destination ROADM in the WSON network.

The MPLS PCE, on receiving the response from the WSON PCE generates a path computation response and sends it to the IP-NCM. It is important that the segments of the path in different network layers are easily identifiable in the final path, and to that end, the MPLS PCE includes *SERVER LAYER INFO* objects [77] in the ERO to indicate the start and end of the path segment in the WSON network. The final multi-layer ERO therefore contains the start MPLS router address (192.168.8.3) followed by the *SERVER LAYER INFO* object and the computed path in the WSON network. Another *SERVER LAYER INFO* inserted after that indicates the end of the path in the WSON network and is followed by the destination MPLS router address (192.168.8.1).

A basic signaling study was performed to evaluate the performance

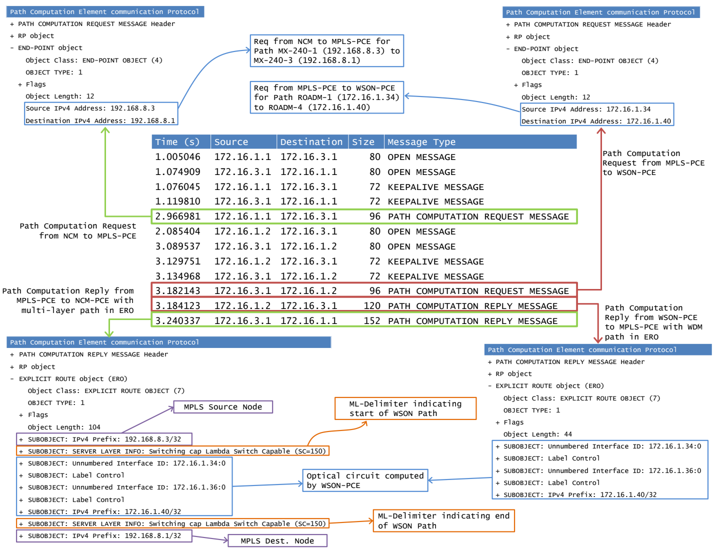


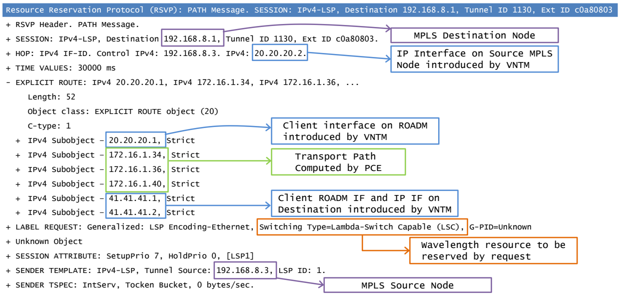
Figure 4.13: Wireshark snapshots of the signaling for path computation between the IP NCM, the MPLS PCE and the WSON PCE [4]

of the path computation setup presented here. In this study, the MPLS and WSON PCE used the Atlanta network topology [66], and the available link capacities of all MPLS links were set to 0 to force multi-layer path computation. The average Round Trip Time (RTT) from the IP-NCM to MPLS PCE and from the MPLS PCE to the WSON PCE was measured using ping traces averaged over 1000 measurements, and was found to be 44.29 ms ($\sigma < 0.15\ ms$). Path computation requests were made between randomly selected endpoints in this topology and the average path computation time was measured as 97.84 ms ($\sigma = 1.02\ ms$). A major component of the total path computation time was the RTTs involved in the signaling between the 1) IP NCM and the MPLS PCE and 2) MPLS PCE and the WSON PCE. The processing time within the MPLS PCE was found to be $\sim 2\ ms$. The WSON PCE used the K-Shortest path algorithm to find feasible paths between WSON nodes, and the First-Fit mechanism was used to identify candidate wavelengths for the optical circuit requested. The increased complexity of the path computation algorithms resulted in a total processing time of $\sim 9\ ms$.

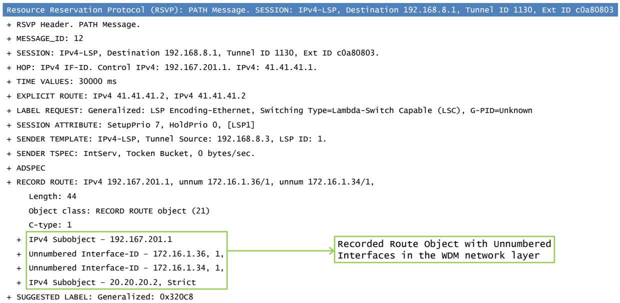
Path Provisioning

The provisioning of the optical circuit computed using inter-layer PCE interaction is initiated by the IP NCM. The IP NCM sends a *TE LINK SUGGESTION* message to the VNTM, which includes the computed multi-layer ERO. The VNTM extracts the optical segments indicated in the multi-layer ERO and initiates provisioning of the optical circuit using the UNI interface. The Wireshark traces of the UNI messages exchanged during the optical circuit provisioning are presented in Fig. 4.14, and highlight some of the critical features required for multi-layer provisioning that are non-trivial.

The path computed by the PCEs indicates the routers and the corresponding ROADMS between which an optical circuit should be established, but do not indicate the interfaces that should be employed for the same. The UNI *Path* message trace presented in Fig. 4.14(a) indicates that the session originates at MX240-1 (192.168.8.3) also includes the control plane address of the IP interface that should be used for provisioning the optical circuit. The ERO used in the path message also includes the address of the client interfaces on the source ROADM, and the address of intermediate ROADM hops (and not the unnumbered interface addresses) to be traversed by the optical circuit. At the destination, the ERO also include the control plane address of the client interface on the ROADM and the terminating IP interface.



(a) UNI Path Message from MX240-1



(b) UNI Resv Message from MX240-3.

Figure 4.14: UNI PATH and RESV messages for optical circuit setup in the WSON network [4]

The required information about the available interfaces as the MPLS routers and the corresponding interfaces on the ROADMS must therefore be available in the VNTM in order to successfully generate the ERO required for the UNI session and provision the optical circuit. Based on this ERO, the UNI session attempts to provision a circuit, and the actual route used for the circuit is presented in the UNI *Resv* message trace (Fig. 4.14(b)).

The provisioning time measured on this testbed using the UNI interface was found to be 54 seconds, which is large primarily due to the physical constraints of current ROADM equipment. The ROADMs require significant time to configure the physical cross-connects and stabilize amplification and noise levels on the established optical channel. The stabilization times have already been reduced in the next generation of optical equipment and are bound to decrease with increased use (and consequently demand) for dynamic optical circuit provisioning.

4.5 Summary

The chapter presented the overview of the *first open-source PCE*, that was developed as a part of this thesis. The open-source implementation is a critical component for closing the gap between PCE research and the development of standards and practical solutions. The PCE implementation presented here has already been used to demonstrate the feasibility of some PCE extensions for multi-path computation [72]. The PCE implementation presented has also been used in EU projects ONE [73] and GEYSERS [74]. A new EU project PACE, scheduled to begin at the end of 2013, will also use the open-source PCE as a component to develop standardization requirements and reference models for application of the PCE in a upcoming network architectures, including SDN-capable networks.

This chapter also presented an overview of the *experimental demonstration of PCE-based multi-layer path computation and provisioning*. The implementation and testbed integration demonstrated the capability of PCE-based architectures to perform coordinated path computation and provisioning in multi-layer networks, and highlighted a number of challenges that must be addressed in the context of standards and implementation before such solutions can be deployed in production networks.

The current standards for the PCE do not define any fixed mechanism for the update of the TED, which is a major implementation overhead in actual network deployments. Therefore, a list of possible solutions, as well as network topology description models for the TED should be proposed in order to ensure that third-party PCEs can be easily deployed in different networks. Extensions to the PCE must also address protocol limitations when requesting and performing complex computations, such as the computation of multi-path connections services or protected path computation.

Another major challenge involves the requirements for essential topology information discovery and exchange across multiple network layers. Information such as the IP address of ROADMs connected to routers, available free interfaces on routers and interconnections between router interfaces and client interfaces on ROADMs is not easily discoverable, especially in multi-vendor networks. It is therefore necessary to identify possible candidates that can perform these actions in different network layers, and employ standards such as the Application Layer Traffic Optimization (ALTO) protocol [78] to exchange topology information between the PCEs in the different network layers.

Finally, components such as the VNTM have been proposed in the context of multi-layer networks in numerous proposals, but current standards do not define any fixed architecture or communication interface for the same. In order to ensure that PCEs and VNTM can inter-operate with each other, it is necessary to define one or more standard interfaces for interacting with the VNTM, which could be extensions to the PCEP as presented here or could also be based on upcoming SDN standards such as OpenFlow [79].

5

Multi-Layer Network Orchestration

The PCE-based solution for multi-layer path computation in the previous chapter highlighted a number of challenges with performing multi-layer operations, which can primarily be attributed to the difference in the technologies and the corresponding management practices. As a result, the preferred solution for performing multi-layer network operations should re-use existing management (sub)systems in the different network layers, and orchestrate the operations across multiple network layers.

The significant interest in upcoming SDN frameworks is demonstrating the demand from the network operators to flexibly define network operations in their networks. The SDN solutions, however, are still nascent, and will not replace existing management systems in the near future. It is, however, possible to use the existing management infrastructure in conjunction with third-party management subsystems such as the PCE to perform basic multi-layer network operations. This thesis work proposed the basic design of the ONE adapter, which is a middleware solution developed as part of the EU Project ONE [73] to facilitate programmable multi-layer orchestration using existing network management subsystems. This project is a collaborative effort with five partners namely: TU Braunschweig, Telefonica I+D, ADVA Optical Networking, Seoul National University and UPC, Barcelona.

This chapter presents the challenges associated with management in multi-layer networks, and describes the novel design features of the ONE adapter solution that enable multi-layer multi-vendor network orchestration. The implementation of specific modules and the integration of all modules within the ONE adapter was also carried out as a part of this thesis work, and is presented in this chapter. The integration of the different modules is especially challenging, given that over 30 different tools/libraries/technologies were used by the different modules in the ONE adapter. Finally, the chapter presents the use of the ONE adapter in specific network scenarios. The application scenarios were proposed by Telefonica I+D based on the challenges faced within their network infrastructure, and this chapter highlights two of these application scenarios, namely IP offloading and PCE-based IP service provisioning which have been presented in this thesis.

5.1 Supporting Publications

1. M. Drogon, M. Chamania, A. Jukan, M. Yannuzzi, X. Masip-Bruin, V. Lopez, O. Gonzalez, M. Maciejewski, M. Roth, C. Brunn, J. Altmann, "Towards Automated Interactions between the Internet and the Carrier-Grade Management Ecosystems," **Future Network and Mobile Summit**, 2013.

2. M. Chamania, E. Demirbilek, A. Jukan, X. Masip-Bruin, M. Yannuzzi, "Using BPEL Workflow Processing for Cross-Layer Orchestrations in IP-over-Optical Networks: A Proof of Concept," **IEEE/IFIP Network Operations and Management Symposium (NOMS)**, 2012.

3. M. Maciejewski, C. Brunn, A. Martinez, X. Masip-Bruin, M. Yannuzzi, W. Ramirez, M. Chamania, A. Jukan, J. Altmann, M. Hassan, O. Gonzalez de Dios, F. Munoz del Nuevo, "Archi-

tectural design of the management adapter," **ONE Project Deliverable D2.2**, 2012.

4. M. Yannuzzi, W. Ramirez, A. Martinez, E. Marin Tordera, R. Serral-Gracia, X. Masip-Bruin, V. Lopez, O. Gonzalez de Dios, A. Azanon, M. Maciejewski, K. Kulaga, C. Brunn, M. Drogon, M. Chamania, A. Jukan, J. Altmann, M. Hassan, "Final report on functional design of the basic and advanced modules," **ONE Project Deliverable D3.3**, 2013.

5.2 Network Management Challenges

IP and optical transport networks have evolved over the years to serve very specific functions in an operator's ecosystem, which is visible from the management practices and system design employed for each of these systems. The IP network currently supports a large number of network services, including best-effort Internet traffic, VPN services and mobile-backhaul, and the inherent flexibility of a packet network makes it the ideal candidate (from the operator's perspective) for rolling-out new services. As a result, most NMSs for the IP network provide features of topology discovery, inventory management, network monitoring and management of alarms, while device and service configurations are performed by highly-skilled network operators over vendor-specific Command Line Interface (CLI) scripts.

On the other hand, optical transport networks are required to support a fixed portfolio of services with very high requirements on availability of the network as well as the management infrastructure. The fixed portfolio of services allows vendors to employ standards for communication, and configuration operations issued from the NMS are first sent to Element Management Systems (EMS), which are specialized management subsystems responsible for the configuration of the actual network equipment. This hierarchy also allows the same

NMS to operate over a diverse network infrastructure. However, transport NMSs also do not inter-operate across multiple vendors, and as a result management of a multi-vendor transport network presents a significant challenge to network operators.

As a result, IP and optical transport networks are treated as separate administrative and management entities within a network operator, and even basic multi-layer operations such as the provisioning of a new IP link are defined as a management process, which involves manual interactions between human operators responsible for managing the two networks. The administrative separation as well as interoperability issues across multiple vendors presents significant challenges for the development and adoption of integrated control and management approaches for multi-layer network.

Upcoming SDN frameworks can address some of these challenges but are not mature enough to replace traditional network management solutions in the core networks. Current SDN solutions have focused primarily on the configuration of operations such as forwarding and measurement in the network, but capabilities such as security, fault tolerance, inventory and topology discovery have not evolved sufficiently to be deployed in a carrier network. Also, current forwarding and measurement abstractions proposed for IP and native Ethernet in the Openflow standard do not exactly conform to technologies such as WDM, and it is highly likely that the next generation of abstractions for managing WDM and flex-grid infrastructure may differ significantly from the existing Openflow abstractions, making integrated multi-layer network management a challenge even in future SDN-capable networks.

5.2.1 The ONE Adapter Approach

The ONE adapter approach is a solution that allows operators to automate operations across multiple network layers [5]. The ONE adapter is not an all-encompassing solution for multi-layer network management. Instead, it focuses on leveraging the existing management infrastructure to perform basic multi-layer network operations. Some of the features and the requirements from the ONE adapter architecture are:

- *Coordinated multi-layer operation*, which requires the ONE adapter to use the capabilities and management practices in the different network layers for performing operations in those layers, with the ONE adapter responsible for coordinating the execution of operations in these layers.
- *Programmable orchestration* implies the capability of operators to define the sequence of operations across different network layers as *workflows* that can be executed by the ONE adapter.
- *Pro-active operations* implies the capability to automatically initiate operations based on network or external events for truly dynamic network operation.
- *Integration of third-party systems* to provide operators automated integration of upcoming and/or specialized third-party components such as the PCE, network planning and traffic engineering tools and within the existing operational workflows.

5.3 Architecture of the ONE Adapter

The ONE adapter architecture, presented in Fig. 5.1 is a Service Oriented Architecture (SOA), which can be segregated into two primary sections, namely the *core* modules and the *auxiliary* modules. The core modules consist of modules responsible for scheduling and managing the definition and execution of operator-defined workflows in the ONE adapter. The core modules therefore contain the administrative and operation logic for multi-layer operations and interact with the auxiliary modules to facilitate the execution of these operations.

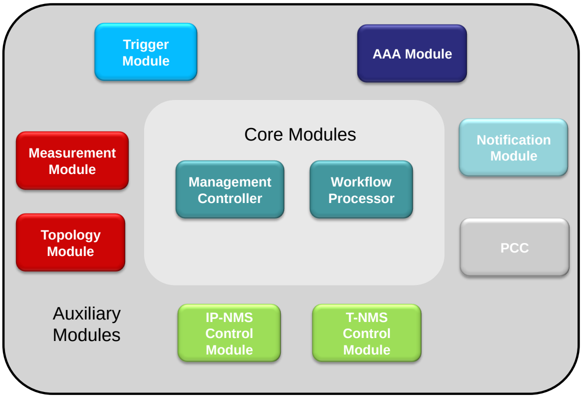


Figure 5.1: Overview of the ONE adapter Architecture [5]

The auxiliary modules interact with external actors inside the operator's management ecosystem in order to gather information required for workflow execution or to perform operations as requested by the workflow. Unlike other middleware solutions, auxiliary modules are not categorized as interfaces to interact with external management actors, but based on specific networking operations that are required in each network layer. Therefore, a single auxiliary module can interact with one or more external actor to perform the necessary network operations. As a result, operational logic is defined as a sequence of networking operations, and not as interactions between external actors making them agnostic to changes in the choice of external actors used.

5.3.1 Typical Execution of a Multi-layer Operation in the ONE Adapter

A typical operation in the ONE adapter starts with the arrival of an *event* which is received by the Trigger module. The event can be a user generated event or an automated event generated by the network or any other external system. The Trigger module processes the event, and if relevant, generates a *trigger* which is sent to the ONE adapter core for initiating an operation. The trigger is received by the Management controller, which is responsible for authorization of incoming triggers, scheduling of operations and performing basic network analytics on incoming network events. To initiate the execution of a process, the Management Controller sends the *trigger* to the workflow processor, which contains the definition of operator defined workflows. The Workflow processor chooses the appropriate workflow for the trigger and initiates execution of the same. Each workflow is defined using basic logical constructs and a series of operation requests (calls) to auxiliary modules, which in turn are responsible for interacting with the external management actors to perform the necessary configurations. The completion of the workflow execution indicates the end of an operation within the ONE adapter, and notifications of the same can be sent to user or other end-systems over different means via the notification module.

5.3.2 ONE Core Modules

Fig. 5.1 indicates two core modules of the ONE adapter, namely the Management controller and the Workflow Processor.

Workflow Processor

The Workflow processor (developed at TU Braunschweig as a part of this thesis) is responsible for the execution of workflows based on incoming triggers from the management controller. The architecture of the workflow processor is presented in Fig. 5.2, and highlights the three major components of the same.

- The *Workflow Database* contains workflow definitions as generated by the operator. Workflows in the ONE adapter are

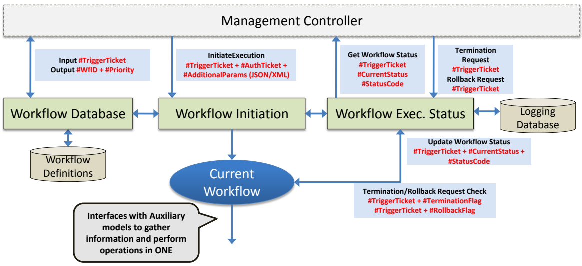


Figure 5.2: Architecture of the Workflow Processor [6]

defined as SOAP based web services, and the workflow definition in this context is stored as the Web Service Description Language (WSDL) specifying the interface to initiate the workflow and the End-point Reference (EPR) to the location of the workflow. The Workflow database also contains the mapping between a specific trigger type and the corresponding workflow. Within the ONE adapter, triggers are always mapped to unique workflow ($N : 1$ mapping).

- The *Workflow Initiation* service receives the request to initiate execution of a trigger from the management controller, extracts the corresponding workflow definition from the Workflow database, and initiates the execution of the workflow. The service is also responsible for validating and using the parameters within the Trigger to generate a web service request to initiate the workflow.

- The *Workflow Status* service provides interfaces for the interaction between the Management controller and a workflow that is being executed. Here, the workflow notifies the Workflow status service of the current status of the workflow which is forwarded to the Management controller. The Management controller can also send a request to *rollback* or *terminate* a workflow to the Workflow status service. The workflow, after every operation, requests the Workflow status service to see if a rollback or termination request has been issued by the management controller, in which case the workflow performs the necessary operations.

Management Controller

The management controller is the central entity responsible for managing and scheduling the incoming triggers for execution in the workflow processor. The primary operations of the management controller include

- *Authorization* of incoming triggers to ensure that only authorized users or network entities are allows to initiate an operation

- *Analytics* to check and apply policies on incoming network events in order to determine if a valid trigger should be executed. For example, triggers generated during a standard maintenance procedure would be analyzed to determine if they correspond to network events associated with the same, and would therefore not initiate execution of additional new workflows. The Analytics feature also defines the priority of a workflow: for example, a trigger for an operation attempting recovery from a failure should be much higher that a standard provisioning operation issued by an operator.

- *Scheduling and Initiation of workflow execution* using the priorities generated by the Analytics function. The scheduling function maintains priority-based process queues and initiates the processing of a trigger by invoking the Workflow initiation service in the Workflow processor.

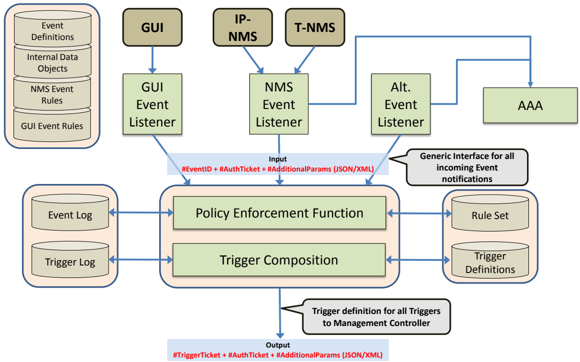


Figure 5.3: Architecture of the Trigger Module [6]

Other than these functions, the management controller also provides services for the operation and administration of the ONE adapter, including services to gather logging information and checking for the operational status of the different modules. The Management controller is also responsible for generating alarms in case any component within the ONE adapter is not functioning properly.

5.3.3 ONE Auxiliary Modules

Fig. 5.1 outlines the primary auxiliary modules of the ONE adapter. Each of the module is an independent service responsible for providing necessary interactions with external actors in the ONE adapter.

Trigger Module

The Trigger module was developed as a part of this thesis work, and is responsible for the generation of *triggers* for invoking operations in the ONE adapter. The architecture of the Trigger module is presented in Fig. 5.3.

The trigger module is designed to receive network events from numerous possible external actors, including a Graphical User Interface (GUI) and external NMSs among others. Dedicated modules are developed to *listen* for network events generated using different protocols, and each module is responsible for providing user or device authentication for the event or the external actor using the Authentication, Authorization and Accounting (AAA) module.

Once the incoming event is validated, JavaScript Object Notation (JSON) based event definitions are generated by the event listeners, and are sent to the policy enforcement function. The use of JSON implies that the interface to the policy enforcement function is the same for all modules, and this enables the easy integration of new event listeners in the architecture.

The policy enforcement function logs the event, and uses a policy rule set to check if a trigger should be generated using this event. A trigger may be generated based on the arrival of a single event (such as a user request generated from the GUI) or can use historical event data to generate a trigger (e.g. multiple alarms notifications). The policy defines the mapping of event data into a trigger. The trigger is composed using the policy definition and is validated against trigger definitions to ensure that the policy was operating correctly. Once this is complete, the trigger is converted into its corresponding JSON notation, logged in a database and is sent to the Management controller. The use of JSON again means that the interface at the Management controller is not extended by the inclusion of a new trigger. Infact, in the implementation, the parameters of the trigger are designed to map to the input of the corresponding workflows, and therefore new triggers and workflows can be introduced at runtime without any change in the implementation of the modules in the ONE adapter.

The trigger module is a critical component within the ONE adapter architecture as it facilitates the initiation of operations from a diverse set of events (operator- or network-generated) and is therefore critical in enabling the ONE adapter to perform automated policy-based operations in a multi-layer network.

Authentication, Authorization and Accounting (AAA) Module

The AAA module provides an interface for the modules within the ONE adapter for user or device authentication, authorization of users or devices to initiate the corresponding workflows, and the accounting of operation executions. The AAA module can provide this service internally or can integrate with external AAA servers using existing standards such as RADIUS [80] or Diameter [81]. AAA is a critical part of existing management ecosystems, and the AAA module within the ONE adapter can also be used to interface with existing AAA systems to perform the necessary authentication and authorization operations with existing AAA servers in different network layers before initiating operations in those layers from the ONE adapter.

IP-NMS Control Module

The IP-NMS control module is responsible for performing configuration operations on network devices in the IP network. The module exposes fixed interfaces for specific operations to the workflow processor, and supports configuration over a number of external actors including

- Invoking configuration operations using interfaces exposed by an IP NMS
- Directly configuring network devices over proprietary CLI or over NETCONF [82].

One of the novel features of the ONE adapter includes the capability to translate commands in a standard syntax to a vendor-specific syntax using a service termed the *Ontology Mapper* [6]. The

Ontology Mapper uses ontology definitions of the various processes performed in the IP NMS and generates relationships to process ontology for different vendors. Using these definitions, commands issued by the workflow processor (and therefore based on the standard syntax) can be converted to the protocol specific syntax for different network vendors. While the ontology-based syntax translation can be used for addressing multi-vendor integration in all modules across ONE, it has been demonstrated especially in conjunction with the ONE adapter, and can convert generic requests for configuration operations to vendor-specific CLI syntax for Juniper, Cisco and Linux-based routers.

The architecture proposed also allows for integration of upcoming SDN standards and has demonstrated the capability to configure forwarding on routers using Openflow [79].

T-NMS Control Module

The T-NMS control module in the ONE adapter is responsible for managing optical service activation and decommissioning by communicating with the transport NMS. Unlike the IP NMS control module, which is required to support a large number of mechanisms to interact with external modules, the T-NMS control module uses the standard interfaces exposed by the T-NMS to perform these operations. However, the design of the T-NMS control module, similar to the IP-NMS control module, allows it to use different mechanisms to configure optical infrastructure. As a demonstration of the same, the T-NMS control module demonstrated the configuration of an optical mesh consisting of ADVA FSP3000 ROADMs using Openflow.

Measurement Module

The measurement module interacts with the network devices or the NMSs in the IP and the optical transport networks to gather measurement statistics from the network. The module can be configured to provide current and historical data, either directly from the external actors if available, or by actively collecting and storing this data in internal databases. This feature is especially useful when performing complex network operations such as traffic engineering. The Measurement module can also be configured to generate events for the Trigger module based on measurements gathered from the network. For example, the measurement module can be configured to generate events in case the capacity utilization of network links increases beyond a specified threshold, which can in turn trigger the ONE adapter to perform traffic offloading.

Topology Module

As the name suggests, the Topology module uses different mechanisms, including interactions with the control plane, SNMP-based discovery and interaction with the NMSs to generate the multi-layer network topology. This information is used by the modules within ONE and can also be exposed as a service to provide multi-layer topology information to external entities (e.g. the PCE TED).

A standard topology representation is critical for the ability to define workflows that can be re-used across different technologies, and the Topology module provides mechanisms to identify and convert addressing across end-technologies used. For example, interface addressing from the point of view of the control plane and the NMS for optical devices is significantly different, and the topology module provides services to convert between the two addressing schemes in order to allow the configuration of the optical network using both the control plane and the NMS.

PCE Module

The PCE module (developed as a part of this thesis) demonstrates the capability of the ONE adapter to integrate with third-party management subsystems. The PCE module exposes a fixed interface to the workflow processor to request path computation, and uses the open-source PCE implementation presented in the previous chapter to initiate a PCEP session with an external PCE. Once the session is initiated, the PCE module generates a PCEP path computation request using the parameters received from the workflow processor, and requests, and forwards it to the PCE server. Upon receiving a response, the PCE module parses the PCEP path computation response message to generate a response to the workflow processor.

The ability to integrate third-party systems in an automated fashion for multi-layer orchestration is one of the major advantages of the ONE adapter, which is not addressed by most middleware solutions. Similar integration with modules performing complex computation such as traffic offloading and multi-layer restoration have also been demonstrated with the ONE adapter.

Notification Module

The Notification module (developed as a part of this thesis) is responsible for providing feedback to external systems or to the network operator. The notification module receives messages from the management controller's logging service and from the workflow processor, and based on policies defined within the notification module, can generate notifications in a number of formats including emails, SMSs or notifications to the GUI.

5.3.4 Implementation Features

The integration of different modules of the ONE adapter was carried out at TU Braunschweig as a part of this thesis. The ONE adapter

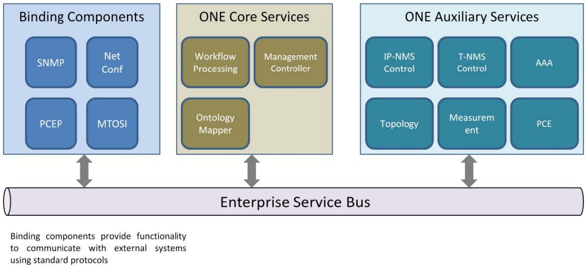


Figure 5.4: Communication between the ONE adapter modules using the Enterprise Service Bus architecture [5]

follows a Service Oriented Architecture (SOA) based approach, and all modules expose Simple Object Access Protocol (SOAP) [83] web service interfaces. Internal communication between the modules is performed over the Enterprise Service Bus (ESB) architecture, which acts as a communication channel as shown in Fig. 5.4.

Each module exposes operations as web services, which are described using WSDL and an end-point location. Actual service locations are registered with the ESB, which advertises an endpoint on the ESB for other modules to *consume* the service. As all consumers request a service over the ESB, this mechanism facilitates service migration and load-balancing between different instances of a service. The ESB can also be used for implementing access control policies and perform message normalization to support integration of different web-service standards.

The ESB architecture also provides numerous binding components that can be used to facilitate communication with external systems. A binding component allows the translation of a web service request to a request on a specific protocol such as PCE, SNMP, etc. and enables the easy integration of interfaces to new external subsystems in the ONE adapter architecture.

In the architecture, the ESB represents a single module over which all communication is routed in the network, and can be a potential bottleneck. However, the architecture has been demonstrated to be scalable, and the WSO2 ESB used in the ONE adapter implementation is currently being used by numerous large e-commerce websites to manage their internal communication [84].

5.3.5 Examples of Multi-layer Orchestration with the ONE Adapter

The ONE adapter architecture has demonstrated the capability to perform a number of multi-layer coordination operations. The specific multi-layer operations that are employed by the ONE adapter have been proposed by Telefonica I+D based on the internal need for these operations and have been outlined in [6], and implementations of these workflows have been developed as a part of this thesis.

These scenarios include basic multi-layer operations such as the provisioning of an IP link, which is developed to demonstrate the capability of the ONE adapter to automate interactions between the management subsystems in the different network layers. The scenarios also include novel operations that are especially critical for network operators such as post-failure recovery. Most networks today are protected in both the IP and the optical network layer, and in case of a failure, the management systems in either of the two networks respond to recover from the failure. However, traffic after the recovery of a failure is not protected, which forces network operators to repair network faults quickly. The post-failure recovery is initiated after a network recovers from a failure, and generates a multi-layer backup for the recovered service, which in turn reduces the time constraint and consequently the cost involved on repairing failures.

Two scenarios, namely IP traffic offloading and multi-layer service provisioning are based on the work done in this thesis, and the thesis presents a brief description of the operation of the ONE adapter for executing these scenarios.

IP Traffic Offloading

The IP traffic offloading workflow is based upon the principle of the Optical Bypass presented in this thesis. In this scenario, the Measurement module monitors traffic on a link and in case the capacity utilization of a link exceeds a pre-defined threshold, the measurement module generates an event, which subsequently results in the provisioning of an Optical Bypass.

An example of this scenario is shown in Fig. 5.5. In this example, traffic from the traffic generator is routed over the initial IP network topology via $MX240-1 \rightarrow MX240-2 \rightarrow MX240-3$ to the traffic sink. Traffic is monitored on the existing links, and in case the traffic on these links increases beyond a specified threshold, an event is generated and sent to the ONE adapter. After the typical processing of the event in the trigger module and the management controller, the offloading workflow is initiated. This workflow computes the location and placement of the optical bypass, and in this scenario, creates an optical circuit from $MX240-1$ to $MX240-3$ (red line) and then configures a new IP link using interface addresses outside the IP routing protocol subnet. After the link is created, a static routing rule is configured on $MX240-1$ to re-route traffic to the traffic sink over the new link from $MX240-1$ to $MX240-3$.

The dynamic traffic offloading workflow demonstrates the capability of the ONE adapter to perform proactive operations based on events in the network. The results from this demonstration have also been reported in [26]. The reported scenario also describes an event to decommission the established dynamic optical circuit. In the setup, once the optical bypass is established, the Measurement module is configured to monitor the traffic on the optical bypass, and generate an event when the traffic on this link decreases beyond

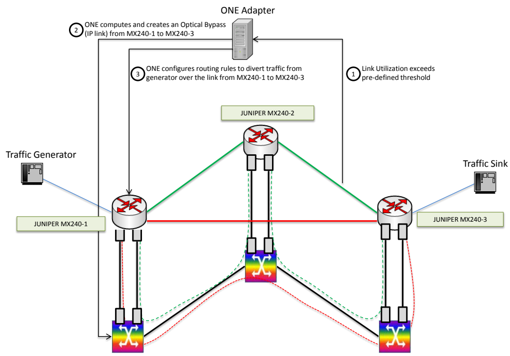


Figure 5.5: Example of IP Traffic offloading using the ONE adapter

a specified *lower* threshold. If this threshold is violated, the ONE adapter triggers a workflow which evaluates whether the decommissioning of the optical bypass is feasible, and if so, first removes the static routing rule at $MX240-1$, so that traffic to the sink is routed back over the original path $MX240-1 \rightarrow MX240-2 \rightarrow MX240-3$ and then decommissions the optical bypass.

MPLS Service Provisioning

This scenario demonstrates the capability of the ONE adapter to provision an MPLS service that involves the configuration of a multi-layer path. This particular scenario is developed based on the PCE-based multi-layer path computation and provisioning demonstration presented in the previous chapter.

In this scenario, the user generates a request to provision an MPLS service, and the ONE adapter uses the PCE module to interact with an MPLS PCE to compute a path for this service. If the path is found in the MPLS network, the ONE adapter configures this path using the IP NMS control module. However, in case the MPLS PCE cannot compute a path, it communicates with the WSON PCE (as described in the previous section) and computes a multi-layer path.

If the ONE adapter receives a multi-layer path, it first extracts the path segment in the optical transport network, and requests the T-NMS to setup an optical circuit in the optical transport network. Once the circuit is established, the ONE adapter generates IP interface configuration and requests the IP-NMS to configure the IP interfaces at the circuit endpoints in order to establish an IP link. Once the IP link has been established, the ONE adapter then modifies the ERO by removing the entries for the route in the WSON network, and then requests the IP NMS control module to configure the MPLS service in the IP network.

In this example, the ONE adapter performs the operations of the IP NCM and the VNTM as described in the previous section. Another configuration of the setup also demonstrates the capability to populate the MPLS PCE TED using the Topology module in the ONE adapter, which overcomes the challenge of determining the ROADMs that are connected to IP routers, which was an open issue with the standardized PCE-based implementation.

5.4 Summary

The ONE adapter architecture, proposed as a part of this thesis, is a novel solution for coordinating network operations in a multi-layer multi-vendor network scenario. The solution stands out as a non-disruptive approach for performing coordinated network orchestration by re-using existing management infrastructure and integrating upcoming third-party management subsystems such as the PCE.

The two scenarios described in this chapter have been proposed by Telefonica I+D, and demonstrate the demand for IP-optical integration techniques such as Optical Bypass and multi-layer path computation and provisioning that have been proposed and studied in this thesis.

The novel design feature which separates the interface and definition of a management operation from the external actor used to perform the same allows the ONE adapter to also integrate with upcoming SDN solutions. The actual implementation of the ONE adapter demonstrates this capability by supporting the configuration of the IP and optical transport network infrastructure using both Openflow and legacy systems.

The architecture's non-disruptive nature, while making it easy to integrate and deploy in current networks, also limits its capability, as it can only perform operations that are exposed by existing management subsystems. However, the features of fixed layer-based abstractions, integration with third-party systems and network feedback as demonstrated in the ONE adapter are critical functionalities for developing truly programmable multi-layer multi-vendor network management solutions, and have been adopted by upcoming SDN-based frameworks such as ANBO [19] and OpenDaylight [18].

6

Conclusion

This thesis highlighted numerous assumptions, and consequent limitations, of existing modeling frameworks in their ability to accurately compute the effects of dynamic optical circuits on IP routing. A novel ILP-based framework was proposed in this thesis that can address many of the shortcomings of existing frameworks. Using a new optimization objective proposed in this thesis, the model can compute the introduction of new dynamic optical circuits as well as the decommissioning of existing dynamic optical circuits, which has not been studied to date. The thesis presented constraints for modeling traditional IP routing, including novel constraints for enforcing *destination-based forwarding* and *routing re-convergence* in the context of SPF routing.

Significant re-configuration of IP routing and the inability to accurately evaluate the location and capacity of dynamic optical circuits under unknown IP traffic matrix conditions were identified as two of the major challenges with IP-optical integration. To this end, the thesis proposed the new Optical Bypass technique that was designed to minimize the effect of introducing dynamic optical circuits on IP routing. A novel formulation for evaluating the ILP model under unknown IP traffic matrix conditions was also proposed. Traditional traffic estimation-based techniques can significantly under-estimate

IP link loads, making the computed solution unreliable. The proposed solution uses easily available traffic measurements in a fashion that ensures that IP link loads are not under-estimated and provides guaranteed upper bounds on the required dynamic optical circuit capacity, which is novel.

A comprehensive numerical analysis, presented in this thesis, demonstrated that the proposed Optical Bypass technique requires *lesser optical capacity* and leads to *fewer routing changes* in the IP network as compared to the traditional SPF schemes. The Optical Bypass technique was also ideally suited for computation under unknown traffic matrix conditions, with results indicating the ability of the model to compute near-optimal solutions under unknown traffic matrix conditions. These features, coupled with its significantly lower computational complexity, make the Optical Bypass scheme ideal for application in production networks.

This thesis outlined management challenges with performing multi-layer operations, and identified the use of specialized management subsystems as a critical component for facilitating multi-layer operations. To this end, the *first open-source PCE* was developed in this thesis work, which is a critical component for constrained path computation and provisioning in multi-layer networks. The thesis work (in collaboration with Telefonica I+D) also demonstrated the experimental validation of PCE-based multi-layer path computation and provisioning in IP-optical networks .

The thesis proposed the basic architecture of the ONE adapter, which is a innovative middleware solution to facilitate multi-layer multi-vendor network orchestration. The ONE adapter orchestrates multi-layer network operations using the capabilities of existing NMSs and third-party management subsystems. The novel architecture of the ONE adapter also allows for integration of upcoming SDN solutions, which is demonstrated by the capability of the solution to use Openflow as well as legacy NMSs to perform configurations in the

IP and optical transport networks.

Two of the application scenarios studied in the EU project ONE, namely IP traffic offloading with the Optical Bypass, and multi-layer PCE-based MPLS service provisioning were based on solutions developed and presented in this thesis. These scenarios were proposed by Telefonica I+D based on the requirements and challenges within their networks, indicating a need for the proposed IP-optical solutions in the near future.

Significant work, however, is required in terms of modeling and management architectures to facilitate better IP-optical integration. The modeling of IP-optical networks presented in this thesis assumed knowledge about the available resources (e.g. IP interfaces) that can be used to deploy dynamic optical circuits. However, the associated network planning problem, which addresses the issue of *where* should additional resources such as IP interfaces be installed in the network, also needs be addressed in order to evaluate the long-term economic benefit of the proposed solutions. Temporal issues pertaining to the use of dynamic optical circuits, such as the estimation of the expected duration of a link or router overloading condition and the frequency of dynamic optical circuit provisioning and decommissioning, also pose an important research and management challenge for IP-optical integration.

It is also critical to develop programmable management solutions for IP-optical networks, which also take into account the emerging SDN trend. Increased dynamism and multi-layer coordination will increase the complexity of multi-layer network management, and new management architectures must therefore evolve to use a distributed software framework. A truly distributed software architecture would not only lower the threshold for developing and deploying new management *applications* that can perform specific multi-layer operations, but will also allow the re-use of specialized management subsystems such as the PCE. Future management applications must also embrace and incorporate the use of primitive abstractions for defining network operations, such as those proposed for forwarding in Openflow, and path definitions in PCE. The use of basic abstractions allow management applications developed using them to control a large spectrum of existing an possibly new network technologies, which is critical.

Bibliography

[1] M. Chamania and A. Jukan, "A comparative analysis of the effects of dynamic optical circuit provisioning on IP routing," *Networking, IEEE/ACM Transactions on*, 2013.

[2] The lambdastation project. [Online]. Available: http://www.lambdastation.org

[3] M. Chamania, M. Drogon, and A. Jukan, "An open-source path computation element (PCE) emulator: Design, implementation, and performance," *Lightwave Technology, Journal of*, vol. 30, no. 4, pp. 414–426, 2012.

[4] O. Gonzalez de Dios, V. Lopez, M. Cuaresma, F. Munoz, M. Chamania, and A. Jukan, "Coordinated computation and setup of multi-layer paths via inter-layer PCE communication: Standards, interoperability and deployment," *Communications Magazine, IEEE*, 2013.

[5] M. Maciejewski, C. Brunn, A. Martinez, X. Masip-Bruin, M. Yannuzzi, W. Ramirez, M. Chamania, A. Jukan, J. Altmann, M. Hassan, O. Gonzalez de Dios, and F. Munoz del Nuevo, "Architectural design of the management adapter," *ONE Project Deliverable D2.2*, April 2012. [Online]. Available: http://ict-one.eu/images/deliverables/one_fp7-infso-ict-258300_d_2_2.pdf

[6] M. Yannuzzi, W. Ramirez, A. Martinez, E. Marin Tordera, R. Serral-Gracia, X. Masip-Bruin, V. Lopez, O. Gonzalez de

Dios, A. Azanon, M. Maciejewski, K. Kulaga, C. Brunn, M. Drogon, M. Chamania, A. Jukan, J. Altmann, and M. Hassan, "Final report on functional design of the basic and advanced modules," *ONE Project Deliverable D3.3*, March 2013. [Online]. Available: http://ict-one.eu/images/deliverables/one_fp7-infso-ict-258300_d_3_3.pdf

[7] M. Jinno, H. Takara, B. Kozicki, Y. Tsukishima, Y. Sone, and S. Matsuoka, "Spectrum-efficient and scalable elastic optical path network: architecture, benefits, and enabling technologies," *Communications Magazine, IEEE*, vol. 47, no. 11, pp. 66–73, 2009.

[8] *Interfaces for the Optical Transport Network (OTN)*, ITU-T Std. Rec. G.709/Y.1331, Feb. 2012.

[9] *Carrier Ethernet Network Architecture Framework Part 2 Ethernet Services Layer - Base Elements*, MEF Std. MEF 12.1, Apr. 2010.

[10] ESnet virtual circuits (OSCARS. [Online]. Available: http://www.es.net/services/virtual-circuits-oscars/

[11] Internet2 phoebus. [Online]. Available: http://e2epi.internet2.edu/phoebus.html

[12] GEANT bandwidth on demand launched as production service to national research and education networks. [Online]. Available: http://geant3.archive.geant.net/Media_Centre/News/Pages/BOD-launched-as-a-production-service.aspx

[13] AT&T optical mesh service. [Online]. Available: http://www.business.att.com/wholesale/Service/data-networking-wholesale/long-haul-accesswholesale/optical-mesh-service-wholesale/

[14] Verizon bandwidth on demand (BOD). [Online]. Available: http://www22.verizon.com/wholesale/solutions/solution/bod.html

[15] PTX series packet transport router. [Online]. Available: http://www.juniper.net/us/en/local/pdf/datasheets/1000364-en.pdf

[16] Cisco introduces the cisco 7600 optical services router. [Online]. Available: http://newsroom.cisco.com/dlls/prod_022001b.html

[17] Open networking foundation. [Online]. Available: https://www.opennetworking.org/

[18] Opendaylight: Linux foundation collaborative project. [Online]. Available: http://www.opendaylight.org/

[19] D. King and A. Farrel, "A PCE-based architecture for application-based network operations," *IETF Internet Draft*, July 2013. [Online]. Available: http://tools.ietf.org/html/draft-farrkingel-pce-abno-architecture-05

[20] Cisco Open Network Environment (ONE). [Online]. Available: http://www.cisco.com/web/solutions/trends/open_network_environment/

[21] Internet2 headroom practice. [Online]. Available: https://wiki.internet2.edu/confluence/download/attachments/17383/Internet2+Headroom+Practice+8-14-08.pdf

[22] L. Cheng, J. Ellson, A. Jukan, P. Lamy, and E. Varma, "Network engineering : Control of dynamic link topology in user networks," *Bell Labs Technical Journal*, vol. 8, no. 1, pp. 207–218, 2003. [Online]. Available: http://dx.doi.org/10.1002/bltj.10054

[23] K. Liu, C. Liu, J. Pastor, A. Roy, and J. Wei, "Experimental study of dynamic ip topology reconfiguration in IP/WDM networks," in *Global Telecommunications Conference, 2001. GLOBECOM '01. IEEE*, 2001.

[24] R. Hao, D. Lee, J. Ma, and J. Yang, "Fault management for networks with link state routing protocols," in *Network Operations and Management Symposium, 2004. NOMS 2004. IEEE/IFIP*, vol. 1, April 2004, pp. 103 –116 Vol.1.

[25] J. Moy, "OSPF version 2," *IETF RFC 2328*, April 1998. [Online]. Available: http://tools.ietf.org/html/rfc2328

[26] M. Yannuzzi, V. Lopez, F. Munoz, A. Azanon, O. G. de dios, J. P. F. Palacios, R. Serral, C. Brunn, and X. Masip, "Coordinated operations between IP and transport network management systems in multi-vendor settings," in *Optical Fiber Communication Conference/National Fiber Optic Engineers Conference 2013*. Optical Society of America, 2013.

[27] K. Shiomoto, I. Inoue, and E. Oki, "Multi-layer network operation and management for future carrier backbone networks," in *Global Telecommunications Conference, 2008. IEEE GLOBECOM 2008. IEEE*, December 2008.

[28] T. Miyamura, E. Oki, I. Inoue, and K. Shiomoto, "Enhancing bandwidth on demand service based on virtual network topology control," in *Network Operations and Management Symposium Workshops, 2008. NOMS Workshops 2008. IEEE*, april 2008, pp. 201 –206.

[29] T. Engel, A. Autenrieth, and J.-C. Bischoff, "Packet layer topologies of cost optimized transport networks multi-layer netwok optimization," in *Optical Network Design and Model-*

ing, 2009. ONDM 2009. International Conference on, Feburary 2009, pp. 1 –7.

[30] S. Xu, K. Sezaki, and Y. Tanaka, “A two-stage simulated annealing logical topology reconfiguration in IP over WDM networks,” in *Telecommunications Network Strategy and Planning Symposium. NETWORKS 2004, 11th International*, June 2004, pp. 327 – 332.

[31] K. Zhu and B. Mukherjee, “Traffic grooming in an optical wdm mesh network,” *Selected Areas in Communications, IEEE Journal on*, vol. 20, no. 1, pp. 122–133, 2002.

[32] Y. Huang, J. Heritage, and B. Mukherjee, “Connection provisioning with transmission impairment consideration in optical wdm networks with high-speed channels,” *Lightwave Technology, Journal of*, vol. 23, no. 3, pp. 982–993, 2005.

[33] X. Chen and A. Jukan, “Optimized parallel transmission in otn/wdm networks to support high-speed ethernet with multiple lane distribution,” *Optical Communications and Networking, IEEE/OSA Journal of*, vol. 4, no. 3, pp. 248–258, 2012.

[34] X. Chen, Y. Zhong, and A. Jukan, “Multipath routing in elastic optical networks with distance-adaptive modulation formats,” in *Communications (ICC), 2013 IEEE International Conference on*, May 2013.

[35] K. Papagiannaki, N. Taft, and A. Lakhina, “A distributed approach to measure IP traffic matrices,” in *ACM SIGCOMM conference on Internet measurement*, ser. IMC ’04, 2004, pp. 161–174.

[36] M. Roughan, M. Thorup, and Y. Zhang, “Traffic engineering with estimated traffic matrices,” in *Proceedings of the 3rd ACM*

SIGCOMM conference on Internet measurement, ser. IMC '03. New York, NY, USA: ACM, 2003, pp. 248–258. [Online]. Available: http://doi.acm.org/10.1145/948205.948237

[37] M. Chamania, M. Caria, and A. Jukan, "Achieving IP routing stability with optical bypass," *Optical Switching and Networking*, vol. 7, no. 4, pp. 173 – 184, 2010, selected Papers from the Third International Symposium on Advanced Networks and Telecommunication Systems (ANTS 2009).

[38] A. Soule, A. Nucci, R. L. Cruz, E. Leonardi, and N. Taft, "Estimating dynamic traffic matrices by using viable routing changes," *IEEE/ACM Trans. Netw.*, vol. 15, no. 3, pp. 485–498, Jun. 2007. [Online]. Available: http://dx.doi.org/10.1109/TNET.2007.893227

[39] D. Harrington, D. P., R. Presuhn, and B. Wijnen, "An architecture for describing simple network management protocol (snmp) management frameworks," *IETF RFC 3411*, December 2002. [Online]. Available: http://tools.ietf.org/html/rfc3411

[40] Configuring custom queueing, cisco IoS 12.0,. [Online]. Available: http://www.cisco.com/en/US/docs/ios/12_0/qos/configuration/guide/qccq.html

[41] *Architecture of transport networks based on the synchronous digital hierarchy (SDH)*, ITU-T Std. Rec. G.803, Mar. 2000.

[42] *Network node interface for the synchronous digital hierarchy (SDH)*, ITU-T Std. Rec. G.707/Y.1322, Jan. 2007.

[43] Bidirectional forwarding detection for OSPF. [Online]. Available: http://www.cisco.com/en/US/technologies/tk648/tk365/tk480/technologies_white_paper0900aecd80244005.pdf

[44] Policy based routing. [Online]. Available: http://www.cisco.com/en/US/products/ps6599/products_white_paper09186a00800a4409.shtml

[45] C. Xin, C. Qiao, Y. Ye, and S. Dixit, "A hybrid optical switching approach," in *Global Telecommunications Conference, 2003. GLOBECOM '03. IEEE*, vol. 7, dec. 2003, pp. 3808 – 3812 vol.7.

[46] I. de Miguel, J. C. Gonzalez, T. Koonen, R. Duran, P. Fernandez, and I. T. Monroy, "Polymorphic architectures for optical networks and their seamless evolution towards next generation networks," *Photonic Network Communications*, vol. 8, pp. 177–189, 2004.

[47] M. Chamania, M. Caria, and A. Jukan, "Achieving IP routing stability with optical bypass," in *Advanced Networks and Telecommunication Systems (ANTS), 2009 IEEE 3rd International Symposium on*, dec. 2009, pp. 1–3.

[48] M. Chamania, M. Caria, and A. Jukan, "Effective usage of dynamic circuits for ip routing," in *Communications (ICC), 2010 IEEE International Conference on*, May 2010.

[49] Router-switch.com world's leading cisco supplier,. [Online]. Available: http://www.router-switch.com

[50] Gurobi optimizer 5.0. [Online]. Available: http://www.gurobi.com

[51] B. Mukherjee, *Optical WDM Networks (Optical Networks)*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2006, p. 422.

[52] X. Chen, M. Chamania, and A. Jukan, "On the effectiveness of optical parallel transmission in IP offloading," in *Networks and Optical Communications (NOC), 2012 17th European Conference on*, 2012, pp. 1–6.

[53] A. Bley, “On the approximability of the minimum congestion unsplittable shortest path routing problem,” in *International conference on Integer Programming and Combinatorial Optimization*, ser. IPCO’05, 2005, pp. 97–110.

[54] M. Caria, M. Chamania, and A. Jukan, “A comparative performance study of load adaptive energy saving schemes for IP-over-WDM networks,” *Optical Communications and Networking, IEEE/OSA Journal of*, vol. 4, no. 3, pp. 152–164, 2012.

[55] M. Caria, M. Chamania, and A. Jukan, “Trading IP routing stability for energy efficiency: A case for traffic offloading with optical bypass,” in *Optical Network Design and Modeling (ONDM), 2011 15th International Conference on*, 2011, pp. 1–6.

[56] M. Caria, M. Chamania, and A. Jukan, “To switch on or off: A simple case study on energy efficiency in ip-over-wdm networks,” in *High Performance Switching and Routing (HPSR), 2011 IEEE 12th International Conference on*, 2011, pp. 70–76.

[57] K. Shiomotoi, D. Papadimitriou, J. Le Roux, M. Vigoureux, and D. Brungard, “Requirements for GMPLS-based multi-region and multi-layer networks (MRN/MLN),” *IETF RFC 5212*, July 2008. [Online]. Available: http://tools.ietf.org/html/rfc5212

[58] Cyan cyms multi-layer management system. [Online]. Available: http://cloud.cyaninc.nu-designs.us/solutions/cy-ms

[59] M. Chamania, M. Drogon, and A. Jukan, “Lessons learned from implementing a path computation element (PCE) emulator,” in *Optical Fiber Communication Conference and Exposition (OFC/NFOEC), 2011 and the National Fiber Optic Engineers Conference*, 2011, pp. 1–3.

[60] A. Farrel, J. Vasseur, and J. Ash, "A path computation element-based architecture," *IETF RFC 4655*, August 2006. [Online]. Available: http://tools.ietf.org/html/rfc4655

[61] J. Vasseur and J. L. Le Roux, "Path computation element communication protocol," *IETF RFC 5440*, March 2009. [Online]. Available: http://tools.ietf.org/html/rfc5440

[62] J. Vasseur, R. Zhang, M. Bitar, and J. L. Le Roux, "A backward-recursive pce-based computation (BRPC) procedure to compute shortest constrained inter-domain traffic engineering label switched paths," *IETF RFC 5441*, April 2009. [Online]. Available: http://tools.ietf.org/html/rfc5441

[63] E. Oki, T. Takeda, J. L. Le Roux, and A. Farrel, "Framework for PCE-based inter-layer MPLS and GMPLS traffic engineering," *IETF RFC 5623*, September 2009. [Online]. Available: http://tools.ietf.org/html/rfc5623

[64] Oscars inter-domain controller application programmer interface. [Online]. Available: http://code.google.com/p/oscars-idc/

[65] Java network I/O. [Online]. Available: http://download.oracle.com/javase/1.4.2/docs/api/java/nio/package-summary.html

[66] SNDLib: survivable fixed telecommunication network design. [Online]. Available: http://sndlib.zib.de/

[67] BRITE: Boston university representative internet topology generator. [Online]. Available: http://www.cs.bu.edu/brite/

[68] R. Guerin, A. Orda, and D. Williams, "QoS routing mechanisms and OSPF extensions," in *Global Telecommunications Conference, 1997. GLOBECOM '97., IEEE*, vol. 3, 1997, pp. 1903–1908 vol.3.

[69] J. Suurballe, "Disjoint paths in a network," *Networks*, vol. 14, pp. 125–145, 1974.

[70] Open source PCE. [Online]. Available: http://www.openpce.org/

[71] M. Chamania, O. de Dios, V. Lopez, M. Cuaresma, M. Drogon, A. Jukan, X. Masip-Bruirr, and M. Yannuzzi, "Coordinated computation of multi-layer paths via inter-layer PCE communication: Standards, interoperability and deployment," in *Communications (ICC), 2012 IEEE International Conference on*, 2012, pp. 6926–6931.

[72] X. Chen, Y. Zhong, and A. Jukan, "Multipath routing in path computation element (PCE): Protocol extensions and implementation," in *Network and Optical Communications, IEEE Conference on*, 2013.

[73] ONE: Towards automated interactions between the internet and carrier-grade management ecosystems. [Online]. Available: http://www.ict-one.eu

[74] GEYSERS: Generalised architecture for dynamic infrastructure services. [Online]. Available: http://www.geysers.eu

[75] S. Martinez, V. Lopez, M. Chamania, O. Gonzalez, A. Jukan, and J. Fernandez-Palacios, "Assessing the performance of multilayer path computation algorithms for different PCE architectures," in *Optical Fiber Communication Conference and Exposition and the National Fiber Optic Engineers Conference (OFC/NFOEC)*, 2013, pp. 1–3.

[76] S. Gunreben and F. Rambach, "Assessment and performance evaluation of PCE-based inter-layer traffic engineering," in *Optical Network Design and Modeling, 2008. ONDM 2008. International Conference on*, 2008, pp. 1–6.

[77] E. Oki, J. L. Le Roux, A. Farrel, and F. Zhang, "Extensions to the path computation element communication protocol (PCEP) for inter-layer MPLS and GMPLS traffic engineering,," *IETF Internet Draft*, July 2012. [Online]. Available: http://tools.ietf.org/html/draft-ietf-pce-inter-layer-ext-07

[78] R. Alimi, R. Penno, and Y. Yang, "Alto protocol,," *IETF Internet Draft*, May 2013. [Online]. Available: http://datatracker.ietf.org/doc/draft-ietf-alto-protocol/

[79] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner, "Openflow: Enabling innovation in college networks," *Whitepaper*, March 2008. [Online]. Available: http://www.openflow.org/documents/openflow-wp-latest.pdf

[80] R. C., S. Willens, A. Rubens, and W. Simpson, "Remote authentication dial in user service (RADIUS)," *IETF RFC 2865*, June 2000. [Online]. Available: http://tools.ietf.org/html/rfc2865

[81] P. Calhoun, J. Loughney, E. Guttman, G. Zorn, and J. Arkko, "Diameter base protocol," *IETF RFC 3588*, September 2003. [Online]. Available: http://tools.ietf.org/html/rfc3588

[82] R. Enns, M. Bjorklund, J. Schoenwaelder, and A. Bierman, "Netconf configuration protocol," *IETF RFC 6241*, June 2011. [Online]. Available: http://tools.ietf.org/html/rfc6241

[83] M. Gudgin, M. Hadley, N. Mendelsohn, J. Moreau, H. Nielsen, and A. Karmarkar, "Soap version 1.2 part 1: Messaging framework (second edition)," *W3C Recommendation*, April 2007. [Online]. Available: http://www.w3.org/TR/soap12-part1

[84] WSO2 ESB success stories. [Online]. Available: http://wso2.com/about/customers

List of Symbols

$\mathcal{V}$	Set of routers in the IP network
$\mathcal{E}$	Set of fixed IP links in the network
$\mathcal{G}(\mathcal{V},\mathcal{E})$	Directed graph of the IP network
v_i	Unique router with index i in the IP network topology
e_{ij}	Directed IP link from v_i to v_j
$\mathcal{V}^{\mathcal{O}}$	Set of switches in the optical transport network
$\mathcal{E}^{\mathcal{O}}$	Set of fixed links in the optical transport network
$\mathcal{G}^{\mathcal{O}}(\mathcal{V}^{\mathcal{O}},\mathcal{E}^{\mathcal{O}})$	Directed graph of the optical transport network
v_i^O	Unique switch with index i in the optical transport network topology
e_{ij}^O	Directed optical link from v_i^O to v_j^O
C_{ij}^{OT}	Available capacity on an optical link e_{ij}^O
$RO_{ij}^{xy}(t)$	Positive integer variable indicating the number of circuits of type t from v_x^O to v_y^O use the optical link e_{ij}^O
w_∞	Very large constant link weight used in the formulation for link does not exist to eliminate the option of the same from shortest path routing
$\hat{L}_{ij}$	Boolean constant to indicate if a fixed IP link exists between v_i and v_j

L_{ij}^{old}	Boolean constant to indicate if an IP link exists between v_i and v_j before optical circuits are established and if it is advertised in the routing protocol
$\hat{w}_{ij}$	Pre-defined link weight metric to define the weight of a link from v_i to v_j if it exists
$\mathcal{T}$	Known Set of granularities for all dynamic circuits supported in the Transport Network
C_t^O	Constant indicating the capacity of an optical circuit of type $t \in T$ in the transport network
N	Maximum number of optical circuits between a pair of routers
$\hat{X}_{xy}$	Boolean indicating if one or more dynamic optical circuits were present from v_x to v_y before starting computation
$\hat{X}_{xy}^t$	Integer indicating number of optical circuits of type $t \in T$ from v_x to v_y before starting computation
$IFCost^t$	Normalized Cost of Optical interface of type $t \in T$
$HopCount_{xy}$	Number of edges in the shortest path computed on the base topology in our performance evaluation
$CostPerHop^t$	Cost of circuit of type t per optical link
$Cost_{xy}^t$	Cost of establishing a circuit of type $t \in T$ from v_x to v_y in the transport network
SW_{xy}^t	Known profit for switching off a single dynamic optical circuit of type $t \in T$ from v_x to v_y

H	Average hop-count between $s-d$ pairs in a topology
$\hat{r}_{ij}^{sd}$	Boolean to indicate if the route from v_s to v_d uses e_{ij} before establishment of optical circuits
$\hat{RC}_{sd}$	Route cost from v_s to v_d before establishment of optical circuits
ψ_{xy}^{sd}	Boolean indicator if route from v_s to v_d traverses v_x and v_y before establishment of optical circuits
$\psi_{xy}^{sd}(ij)$	Boolean indicating if route from v_s to v_d traverses v_x and v_y and the segment v_x to v_y traverses link e_{ij}
P	Large positive constant to constrain route change when no shorter path is found
P_∞	Large positive constant to linearize routing re-convergence constraints
α	Link load threshold in the IP network
$\hat{\lambda}_{sd}$	(Known) IP Traffic from v_s to v_d
$LinkLoad_{ij}$	Traffic measured on link e_{ij} before establishment of optical circuits
γ_{ik}^{j}	Virtual Output Queue Traffic measured on v_j incoming from v_i and routed to v_k before establishment of optical circuits
D	Set of linear traffic expressions obtained from link load and VoQ traffic measurements
D_i	Unique Traffic Expression in D
d_{sd}^{i}	Boolean coefficient of λ_{sd} in D_i
B_i	Value of expression D_i

λ_∞	Very large positive constant >> sum of all traffic in the IP network
λ_{sd}^{max}	Upper bound computed on traffic from v_s to v_d using the set of traffic expressions in D
λ_{sd}	Variable indicating IP Traffic from v_s to v_d
X_{xy}^t	Integer variable to indicate the number of dynamic circuits of capacity C_t^O provisioned from v_x to v_y
X_{xy}	Boolean variable to indicate if one or more dynamic optical circuit is established from v_x to v_y
L_{ij}	Boolean variable to indicate if an IP link exists between v_i and v_j after optical circuits are established and is advertised in the routing protocol
r_{ij}^{sd}	Boolean variable to indicate if route from v_s to v_d uses the link from v_i to v_j
$FT_i^d(j)$	Boolean to indicate if forwarding table at v_i has link to v_j as next hop to destination v_d
RC_{sd}	Routing cost for the route from v_s to v_d
w_{ij}	Link weight metric variable to define the weight of a link from v_i to v_j
V_{sd}	Boolean variable used to linearize the absolute value function for routing reconvergence in SPF
Y_{xy}^t	Boolean variable to indicate if X_{xy}^t is greater than the original number of dynamic optical circuits on this link
Z_{xy}^t	Integer variable to linearize the product $X_{xy}^t \cdot Y_{xy}^t$

f_{xy}^{sd}	Boolean variable to indicate if traffic from v_s to v_d uses a bypass from v_x to v_y
f_{xy}^{d}	Boolean variable to indicate if traffic to v_d uses a bypass from v_x to v_y
C_{ij}	Variable to indicate total capacity of IP link e_{ij}
$\hat{C}_{ij}$	Known total capacity of fixed optical circuits installed from v_i to v_j
a_{ij}^{sd}	Boolean variable to indicate the contribution of λ_{sd} for traffic on link from v_i to v_j for formulation in unknown traffic matrix conditions
S_{ij}^{x}	Boolean variable to indicate if traffic on e_{ij} matches the traffic expression $D_x \in D$
S_{ij}	Boolean variable to indicate if no $D_x \in D$ matches the traffic expression on link e_{ij}

Acronyms

MIP	Mixed Integer Programming
SNMP	Simple Network Management Protocol
ER	Explicit Routing
ER-D	Explicit Routing under Destination based Forwarding
SPF	Shortest Path First Routing
BY	Optical Bypass
BY-D	Optical Bypass with Destination-Based Forwarding
SPF-NoTM	Shortest Path First Routing under Unknown Traffic Matrix Conditions
BY-NoTM	Optical Bypass under Unknown Traffic Conditions
BY-D-NoTM	Optical Bypass with Destination-Based Forwarding under Unknown Traffic Conditions
MPLS	Multiprotocol Label Switching
LSP	Label Switched Path
PCE	Path Computation Element
OSPF	Open Shortest Path First
VOQ	Virtual Output Queues
SDH	Synchronous Digital Hierarchy
OTN	Optical Transport Networks
WDM	Wavelength Division Multiplexing
ECMP	Equal Cost Multi-Path routing

ILP	Integer Linear Program
MILP	Mixed Integer Linear Program
QoS	Quality of Service
LHS	Left Hand Side
RHS	Right Hand Side
IP	Internet Protocol
SLA	Service Level Agreement
NMS	Network Management Systems
EMS	Element Management Systems
CLI	Command Line Interface
AAA	Authentication, Authorization and Accounting
IETF	Internet Engineering Task Force
SOA	Service Oriented Architecture
SP	Shortest Path Routing
SPDF	Shortest Path Routing with Destination-based Forwarding
PCEP	Path Computation Element Communication Protocol
TED	Traffic Engineering Database
PCC	Path Computation Client
NAT	Network Address Translation
VPN	Virtual Private Networks
WSON	Wavelength Switched Optical Network
VNTM	Virtual Network Topology Manager
NCM	Network Control and Management
ROADM	Reconfigurable Optical Add Drop Multiplexer
UNI	User-Network Interface
ERO	Explicit Route Object
RTT	Round Trip Time
ALTO	Application Layer Traffic Optimization

SDN	Software Defined Networking
WSDL	Web Service Description Language
EPR	End-point Reference
GUI	Graphical User Interface
JSON	JavaScript Object Notation
SOAP	Simple Object Access Protocol
ESB	Enterprise Service Bus
OSNR	Optical Signal to Noise Ratio
API	Application Programming Interface
REST	Representational state transfer